Transcription Regulation: models for combinatorial regulation and functional specificity

David John Thomas

A thesis submitted in partial fulfilment of the requirements of the University of Sussex for the degree of Doctor of Philosophy

January 2014

Declaration

I hereby declare that the research contained in this thesis, unless otherwise formally indicated within the text, is the original work of the author. The thesis has not been and will not be, submitted in whole or part to another University for the award of any other degree.

Signed: ……..

Dated: ……..

Acknowledgements

My first thanks go to my supervisor Dr Susan Jones for the massive amount of valuable advice, support, and assistance provided over the period of my studies. I especially appreciate the continued support despite the relocation to a different country. Thanks also to my second supervisor, Professor Melanie Newport, for really useful advice and support as well as the "prodding" to stop me procrastinating.

Colleagues and collaborators provided considerable support. Dr Kirsty Flower was great to work with on the EBV genome as were Professor Florian Kern and team with the immunological work. A special mention should go to Dr Chris Finan for an incredible amount of advice and patience on a whole range of bioinformatic topics as well as being a pleasure to work with.

The transitory members of write-up room 2 made coming into work enjoyable and entertaining. Thanks to Chris, Gillian, Helen, Natalie, Natasha and Sophie for the varied conversations.

My family have been a great support, my wife Louise for proofreading and accepting a mature student in the house. My daughters Ellen and Beth should be thanked for not being too embarrassed about their dad going to college and for Photoshop advice.

Final thanks go to the Medical Research Council for the valuable financial support.

Abstract

Gene regulation is controlled by transcription factor proteins that bind to specific DNA sequences, known as transcription factor binding sites (TFBSs). Combinations of transcription factors working, co-operatively in *cis*-regulatory modules (CRMs), play a role in regulating gene expression. Current computational methods for TFBS prediction cannot distinguish between functional and non-functional sites, and predict very large numbers of false positives.

This thesis focuses on the development of a novel computational model, based on artificial neural networks (ANNs), for the identification of functional TFBSs, and the CRMs within which they operate in the human genome. Datasets of 12,239 experimentally verified true positive (TP) TFBSs and 130,199 false positive (FP) TFBSs were extracted using a combination of position weight matrices from the JASPAR database and experimentally verified sites from the Encyclopedia of DNA elements (ENCODE). A number of machine learning algorithms were tested using a range of genetic information including gene expression, nucleosome positioning, DNA methylation states and DNA entropy. The best model, that gave a mean area under the curve under a receiver operator characteristic curve of 0.800, was based on a feedforward ANN using backpropagation.

This model was then used to predict functional TFBSs in a number of gene sets from the human genome. The predictions, combined with experimentally proven TFBSs from ENCODE, were used to investigate combinatorial patterns of TFBSs operating in CRMs. CRM patterns have been analysed in disease-associated genes located in linkage disequilibrium blocks containing SNPs obtained from Genome Wide Association Studies (GWAS).

The potential for the model to make functional TFBS predictions to aid in the annotation of orphan genes of unknown function is discussed. In addition this thesis presents computational work on a number of smaller published studies.

Table of Contents

List of Figures

List of Tables

Abbreviations

ANN	Artificial Neural Networks
API	Applications Program Interface
AUC	Area Under the Curve
ChIP	Chromatin Immunoprecipitation
CMV	Cytomegavirus
CRM	*Cis*-regulatory Module
EBV	Epstein-Barr Virus
EMSA	Electrophoretic Mobility Shift Assay
ENCODE	Encyclopedia Of DNA Elements
ETL	Extraction, Transformation, Load
GO	Gene Ontology
GWAS	Genome Wide Association Study
HAVANA	Human and Vertebrate Analysis and Annotation
HK	Housekeeping (facultative)
HMM	Hidden Markov Model
HPC	High Performance Cluster
ICD	International Statistical Classification of Diseases
IFN-γ	Interferon Gamma
LMA	Levenberg Marquardt Algorithm
NBC	Naïve Bayes Classifier
NPS	Nucleosome Positioning Software

MIAME	Minimum Information About Microarray Experiment
MLP	Multi Layer Perceptron
MLR	Multiple Linear Regression
PFM	Position Frequency Matrix
PIC	Pre-initiation Complex
PWM	Position Weight Matrix
RMS	Root Mean Squared
ROC	Receiver Operating Characteristics
RP	Resilient Propagation
SCG	Scaled Conjugate Gradient
SETS	Sequence, Expression, Temporal, Structural
SNP	Single Nucleotide Polymorphism
SOM	Self Organising Map
SVM	Support Vector Machine
TF	Transcription Factors
TFBS	Transcription Factor Binding Sites
TFFM	Transcription Factor Flexible Models
TNF-α	Tumour Necrosis Factor Alpha
TP	True Positive
TS	Tissue Specific (constitutive)
TSS	Transcription Start Site
UML	Unified Modelling Language
URS	Upstream Repressing Sequences
ZRE	Zta Response Elements

1 Introduction

1.1 Thesis Aims

The first aim of this work is to create a computational method, using machine learning techniques, that integrates information from a variety of data sources to create a novel method of classifying functional and non-functional TFBSs in the human genome.

The second aim is to apply the method to the prediction of combinations of TFBSs working together in *cis*-regulatory modules (CRMs), using gene sets available from the Genome Wide Association Studies (GWAS) repository at the US National Human Genome Research Institute (NHGRI) (Hindorff et al. 2011).

The introductory chapter outlines a number of key areas of gene regulation pertinent to the modelling. This is followed by a discussion of what data is available and how it has been obtained, and an introduction to machine learning modelling.

1.2 Gene Regulation

Gene regulation is controlled in part by proteins known as transcription factors (TFs). TFs bind to DNA sequences at specific binding sites known as Transcription Factor Binding Sites (TFBSs) to activate or repress gene expression. TFBSs are short sequence motifs, usually between 5 and 15 nucleotides in length(Wasserman & Sandelin 2004).

Genes that have strongly correlated mRNA expression profiles have an increased probability of having common TFs (Allocco et al. 2004) (Brown et al. 2007). It has also been shown that genes that have shared functional annotations, core biological processes as defined by the Gene Ontology Consortium (Ashburner et al. 2000), have an increased likelihood of having a shared TF. In Drosophila melanogaster this has been shown to be more important than shared expression profiles (Marco et al. 2009).

The identification of TFBSs using laboratory techniques is time consuming and costly, and hence algorithms have been developed to predict the locations of TFBSs in genomic DNA (Hannenhalli 2008). The identification of genes that share TFs provides evidence for the expressed genes being present in the same or related biochemical pathways. Hence TFBS prediction can be used as a method for mapping gene products to pathways associated with complex disease processes (Vaquerizas et al. 2009).

TFBS functionality is further complicated by the fact that a gene can be co-regulated by multiple transcription factors operating co-operatively by binding as *cis*-regulatory modules (CRMs) (Wasserman & Sandelin 2004) . In addition, the current approaches to TFBS prediction fail to distinguish between functional and non-functional TFBSs. Furthermore, predicting a TFBS in-vitro gives no indication of its in-vivo binding potential (Qiu 2006). This inability to distinguish between functional and non-functional, which results in prediction methods that produce large numbers of false positives, has been termed the "Futility Theorem" (Wasserman & Sandelin 2004). This results in the prediction of greater than 1000 false positives for each true positive.

The regulation of gene expression is intricately controlled by mechanisms that control access to DNA. In eukaryotes, this regulation needs to be fine-tuned to cope with the varied requirements of distinct tissue types; for example, a white blood cell needs to produce antibodies whilst a pancreatic cell needs to synthesize insulin. In addition gene expression is responsive to different conditions, with expression being up or down-regulated or turned on or off depending on the conditions.

Recent projects, especially ENCODE (Becker 2011) have suggested that the genome has more functionality than previously thought, with >= 80% of the human genome estimated to have a specific function (Dunham et al. 2012). This has however been disputed by other studies, both the use of the term "function" and the large variations in genome sizes between organisms (Graur et al. 2013). An additional criticism is the classification of Single Nucleotide Polymorphisms

(SNPs) with observed effects in Genome Wide Association Studies (GWAS) as being indicative of functional DNA (Niu & Jiang 2013).

The regulatory processes that relate to the data used in the modelling (chapters 3,4,5) are presented in the following sections.

1.2.1 The Regulation of Transcription

Regulation of the expression of eukaryotic protein-coding genes can occur at several points although the most common point is thought to be at the point of transcription initiation (Maston et al. 2006). Prior to mRNAs being produced by transcription, proteins attach to specific strands of DNA to form the pre-initiation complex (PIC), see figure 1-1. Although there are no universal promoter elements (Butler & Kadonaga 2002), a mechanism common to the regulation of almost all eukaryotic genes is the formation of the RNA Polymerase II complex that forms part of the PIC. This complex is often found alongside a TATA box and located close to the TSS.

The RNA polymerase II complex forms around the recognition element for the TFIIB transcription factor. Other basal or general transcription factors, TFIIA, TFIID, TF11E, TFIIF, TFIIH are recruited and bound forming the core promoter or the minimal transcription initiation complex.

In addition to this basic polymerase complex, a variety of other mechanisms regulate the spatial and temporal production of mRNAs. Locus control regions containing upstream activating sequences (UAS) including enhancers and upstream repressing sequences (URS) such as silencers can be at large distances from the TSS. Distances of 85kb from the TSS have been reported (Lee & Young 2000), and even effects emanating from other chromosomes have been observed (Miele & Dekker 2008). Transcription factors, either singly or working in CRMs however are predominately found operating nearer the TSS. A study of promoter activity based on ENCODE regions (Cooper et al. 2006) observed most positive regulatory effects to be seen between 50bp and 300bp upstream of the TSS. An increase in negative effects also being seen between 500bp and 1000bp upstream in 55% of genes analysed.

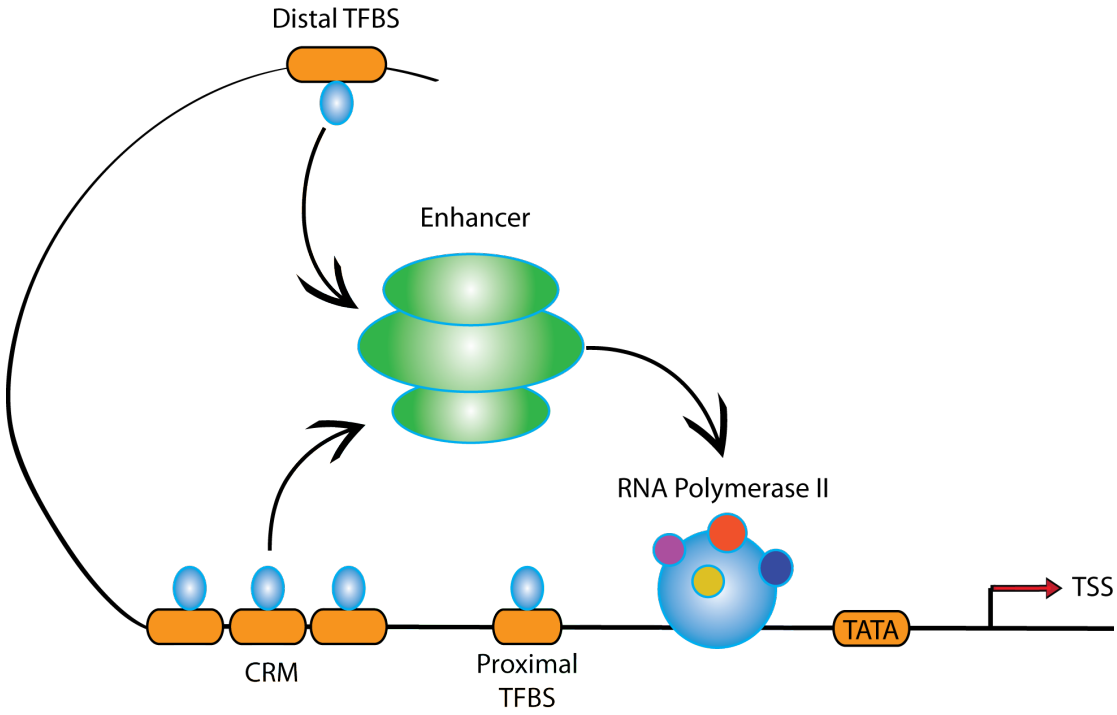

Figure 1-1 Pre-initiation Complex. TFBSs binding with enhancers to the RNA Polymerase II complex prior to transcription.

1.2.2 Chromatin Accessibility

In eukaryotic genomes, DNA is compacted within the nucleus by histone proteins, 147 base pair sequences of DNA that are tightly wrapped around a histone octamer to form a nucleosome. These structures are connected by unwrapped linker DNA of typically 10-50 base pairs in length, producing the "beads-on-a-string" DNA structure (Figure 1-2) (Annunziato 2008). The functionality of TFBSs has been shown to be affected by epigenetic factors such as positioning relative to nucleosomes (Daenen et al. 2008) (Tillo et al. 2010). For TFBSs within the DNA to be accessible to TFs they need to be positioned outside of nucleosome structures or the structures need to be opened by transcriptional machinery. Chromatin remodelling complexes are responsible for the selective reorganization of nucleosomes to permit this access (Jiang & Pugh 2009).

DNAse I hypersensitivity sites are areas of the genome where DNAse I is heavily recruited to cleave the DNA into single strands, and therefore can be used as a marker for open chromatin and hence potential regions for regulatory activity (Elgin 1988). Furthermore, these sites have recently been shown to have sequence specificity (Koohy et al. 2013) leading to potentially more accurate predictions of regulatory locations in the future. This accessibility of the histone tails and the ability to be chemically modified permits epigenetic modifications affecting temporal and spatial expression.

1.2.3 Epigenetic Modifications

The relative access of DNA within nucleosomes and their histone tails therefore have a role in regulating gene expression by regulating the accessibility of TFs to their TFBSs (Segal et al. 2006). Methylation of DNA attracts proteins that increase the deacetylation of nearby histones thereby inhibiting transcription (Tost 2009). Demethylated DNA allows the DNA to remain acetylated and hence more open to transcriptional machinery (Tost 2009), see figure 1-2. However, there are exceptions to this rule, for example, H3K4Me3 has a positive association with gene expression (Schones et al. 2008) (Zhao & Han 2009). In addition, histone modifications have been implicated in cancers via over expression of genes and the silencing of tumour suppressing genes (Zhang et al. 2010).

In addition to histone modifications, sequence specific epigenetic data is also available by looking at the presence of key regulatory elements with CpG Islands, TATA boxes and CAAT boxes providing insight into promoter location and function (Cooper et al. 2006).

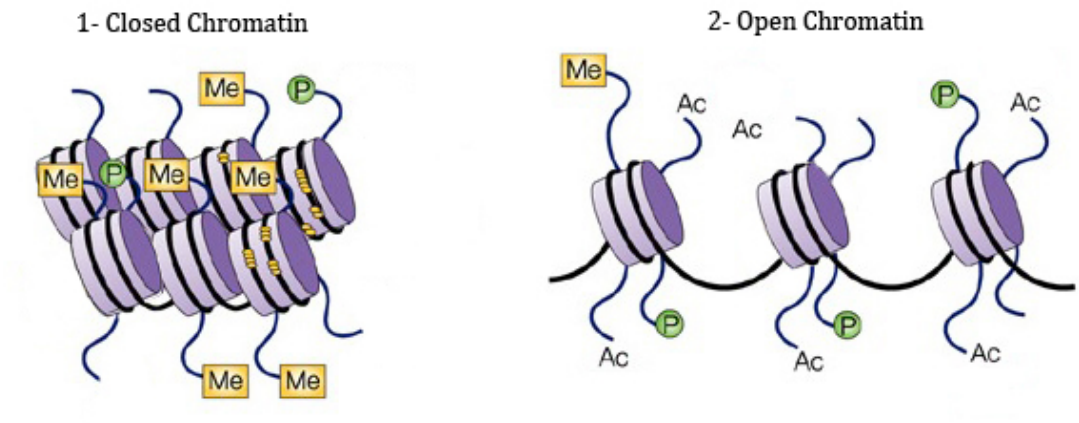

Figure 1-2 Nucleosomes showing 1] closed state, predominately methylated, preventing transcription. 2] open state, predominately acetylated allowing transcription (Figure adapted from Nature Reviews: Drug Discovery April 2002, R Johnstone).

1.3 Data Availability

As discussed in section 1.2, a number of different genomic parameters influence transcriptional regulation and these parameters (and others) are available for the Human genome through a number of public data sources.

The DNA sequences of the human genome assembly are standardised and maintained by the Genome Reference Consortium (Lander et al. 2001), managed by GenBank (Benson et al. 2006) and made available via the Biomart (Smedley et al. 2009) service of Ensembl (Flicek et al. 2010). This data has been used for the extraction of upstream and downstream regions of genes to enable variables to be created for analysis and classification.

The ENCODE (Thomas et al. 2007)project provides data on experimentally proven TFBSs and epigenetic modifications obtained via ChIP-Seq techniques. This technique identifies interactions between DNA and proteins associated with chromatin, including TFs and histones, via chromatin Immunoprecipitation (ChIP). A specific antibody is used to bind to the protein of interest and then these are sequenced and compared to whole genome sequence databases to examine their interactions (Barski & Zhao 2009).

The regulatory build from Ensembl has been compiled by examining results on CD4+ T-cells (Flicek et al. 2013) and provides data on accessibility via DNAse I hypersensitive sites, locations via RNA Polymerase II sites, and a various histone modifications involved in activation, repression and elongation via methylation and acetylation.

ArrayExpress (Parkinson et al. 2007) provides gene expression levels obtained by microarrays in addition to high throughput sequencing experiments. The microarray experiments consist of chromatin-Immunoprecipitation followed by a hybridization to a microarray chip and so is also referred to as ChIP-chip analysis (Ho et al. 2011).

Data pertaining to nucleosome occupancy is based on in silico analysis and consists of algorithms analysing stretches of DNA sequence and predicting the likelihood of positions being part of a nucleosome or linker DNA (Xi et al. 2010).

Each data element plays a role in determining transcriptional regulation and

ultimately gene expression. In order to integrate the very large numbers of these (for example there are 144,680 transcripts in the human genome with a total of 355,570 epigenetic markers in Ensembl GRCh37.12 (Flicek et al. 2010) into a model for the prediction of functional TFBSs, machine learning techniques from the computer sciences need to be implemented.

1.4 Machine Learning Modelling

1.4.1 Motivation

The large number of TFBS false positives seen when using traditional prediction methods such as Position Weight Matrices (see 2.1.1) is inevitable when using the DNA sequence alone. The typical 5-15 base pair length sequences will be observed many times when we examine the human genome with forward and reverse strands of c3.2 billion base pairs (Wasserman & Sandelin 2004). As an example, a typical TFBS for the transcription factor USF1 (Upstream Transcription Factor) has a consensus sequence of CACGTGT. In a random sequence of DNA we should expect to see this sequence once every 4^7 base pairs or 16,384 base pairs. Considering both strands of DNA we would expect to see 390,625 exact matches in the human genome.

Many models have been built to address this problem either by using the sequence alone (Maston et al. 2006; Murakami et al. 2004), or by using phylogenetic data for sequence comparison between species (Håndstad et al. 2011; Bickhart & Liu 2013) or by combining other data such as gene expression (Marco et al. 2009) or nucleosome occupancy predictions (Daenen et al. 2008).

However, to combine data from the multiple sources presented in this thesis is a novel approach. As the approach involves many inputs from different types of biological data resulting in a single output class, the pattern recognition and classification abilities of machine learning have been considered most appropriate.

Machine learning techniques cover a wide range of methods including such disparate methods as statistical (Bayesian) techniques and multi-level artificial neural networks (ANN) (Hopfield 1982). Here we focus on Artificial neural networks (ANNs) which provide a simplistic model of biological neural networks. ANNs are pattern classifiers that take inputs and apply them to one of several output classes and can be thought of as an extension of statistical pattern recognition (Bishop 1995). In all these models, iterations are performed over training sets with the intention of producing a more accurate classification or prediction on a test set via processes of trial and error via learning (Kaelbling et al. 1996).

1.4.2 Types of ANN

There are two categories of ANNs comprising of those that use supervised learning and those that use unsupervised learning techniques. Supervised learning is possible if the aim is to predict an event. In this case a training set is used to compare predicted with actual results. Values at synapses points are normally initially assigned random values, which are improved, by amendment and examination to see which ones work best during the training phase of the

model.

The key types of supervised ANN include a) Support vector machine (SVM), a binary linear classifier, these can be well suited to comparing and classifying two states of a dependent variable (Hsu et al. 2010) and b) feedforward neural networks, these are ANNs that provide scores and weights feeding forward through the layers of the model (Montana 1989). Although there are differences in the way these techniques are applied, the accuracy of their results tends to be similar (Romero & Toppo 2007).

A distinct method of training ANNs are Genetic Algorithms (Whitley 1994), these seek to model evolution via natural selection. In this method, each observation in the training set is a member of a population, their variables can be thought of as genes on a single chromosome, the genes can be expressed at various levels or not at all. Initially assigned random weights as with the other models, they mimic natural selection as a method of amending their values. At the end of an iteration, the fitness of population is checked by how well they predict the outcome and a subsection of the population (say the 50% most accurate predictions) "mate" to produce the next generation. The mating process involves crossing over of some variables; typically two cut points are created to allow three sections of "genes" to be swapped over to mimic recombination. Mutations are also introduced by randomly amending some weights by a small percentage. This process can be repeated until a satisfactory result is achieved.

Unsupervised learning is a form of clustering, the technique attempts to find structures and patterns in data that does not have a specific event to predict. For example, if insufficient data is available on experimentally verified TFBSs, data can still be presented to an unsupervised technique to create useful models. Common unsupervised ANN techniques include the Self-Organising Map (SOM), also known as Kohonen Neural Networks (Kohonen 1990), and Hidden Markov Models (HMM), a dynamic Bayesian model (Eddy 1998).

1.4.3 How a machine learns

In ANNs, perceptrons are the equivalent of neurons; they receive inputs and can react by firing if a threshold is reached. The arrangement of perceptrons into layers, combined with methods of amending weights or delta scores are the methods that affect the complexity of the models (Riedmiller 1994). The layers consist of an input layer, a number of hidden layers (usually one but more can be used) and an output layer consists of the results of the model (see Figure 1-3). The number of perceptrons in the hidden layer is a key parameter that has significant effects on the results. Too few and the model will underperform by not being able to make the most of the input data; too many and the risk of over fitting and hence just describing the current data will rise (Tetko et al. 1995). The decision on whether a neuron will fire is based on an activation function being applied to the data. As a large number of inputs will enter the model, a sigmoidal function is typically used (Bishop 1995).

Iterations are performed until a required result is reached or a certain number of loops have been tried. At the end of an iteration, results are compared to the known results of the training set and errors calculated by means of Root Mean Squared (RMS). Weights are then amended if incorrect decisions have been made. This iterative training process is required by almost all models to allow them to learn to make more appropriate decisions, the goal of ANNs generally being to reduce error thereby producing the most accurate model. Weights are amended by applying a factor to them that can be positive or negative based on the error. This factor is also affected by the concept of momentum in which, if results are improving, the factor increases whilst the opposite happens if the improvements in model performance is decreasing. This also helps to prevent a common issue with ANNs, that of local minima (Basheer & Hajmeer 2000).

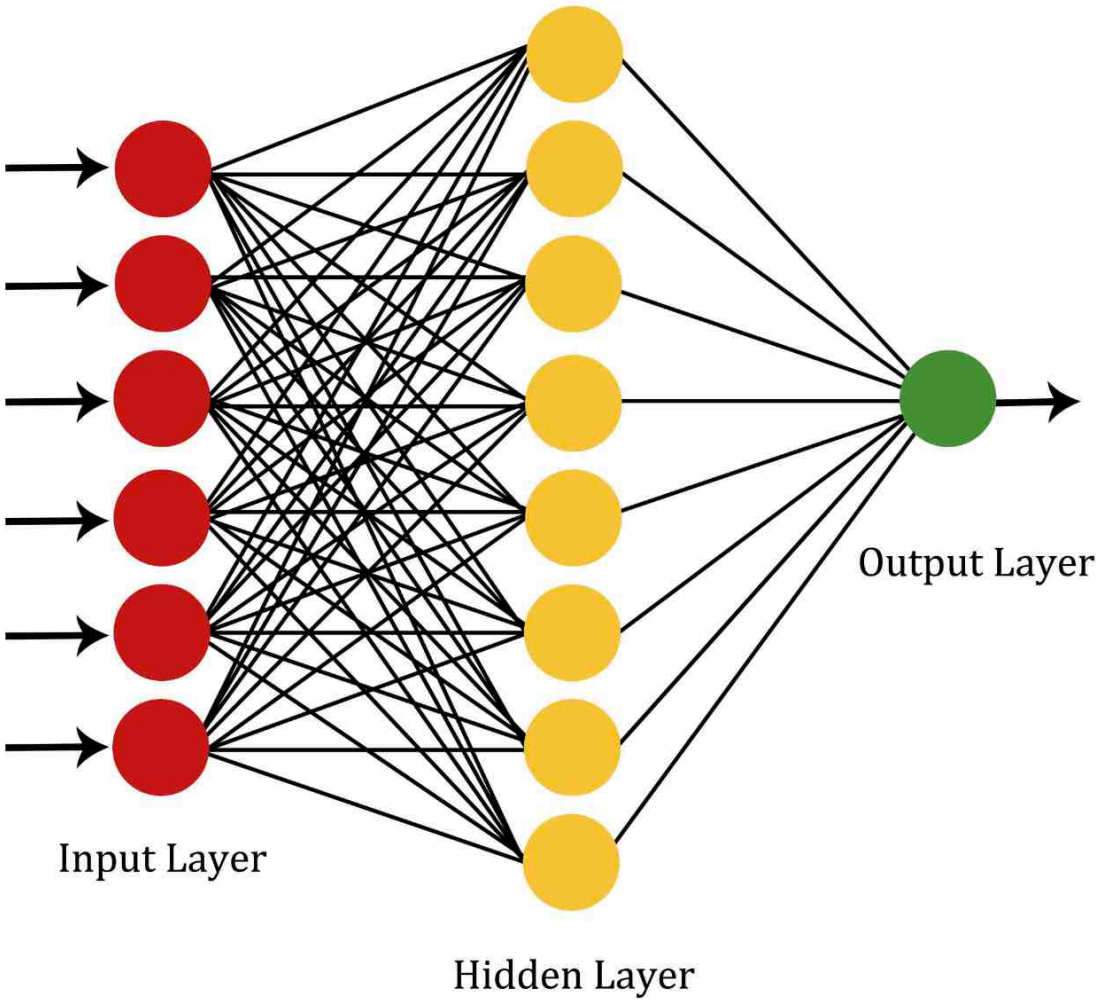

Figure 1- 3 Multi layer Perceptron showing three layers and their connections.

1.4.4 ANN Modelling Summary

At the outset of this project it was possible that insufficient data on experimentally proven TFBSs would be available to perform supervised learning techniques. Unsupervised techniques were therefore investigated despite their being less suitable to predicting true and false positives. However, as the ENCODE project moved from pilot to main project and started to release more

data, initially in 2010 with the major release in 2012 (Bernstein et al. 2012), it became possible to use supervised ANNs for this work.

The combination of availability of techniques in the chosen framework (see Chapter 4) and the potential similar results between SVMs and feedforward models (Romero & Toppo 2007) resulted in various types of feedforward ANNs together with the distinctive training method of Genetic Algorithms being used for the testing phase of the predictive modelling.

1.5 Summary

The aim of this thesis is to develop a new model for the prediction of functional TFBSs and their organisation into CRMs. For this, information from a variety of data sources including predicted TFBS sequences (Portales-Casamar et al. 2010), experimentally verified TFBSs (Bernstein et al. 2012), nucleosome positioning sequence (NPS) predictions (Xi et al. 2010), gene expression data from ArrayExpress (Parkinson et al. 2007) and epigenetic data from Ensembl (Flicek et al. 2010), have been integrated (Figure 1-4).

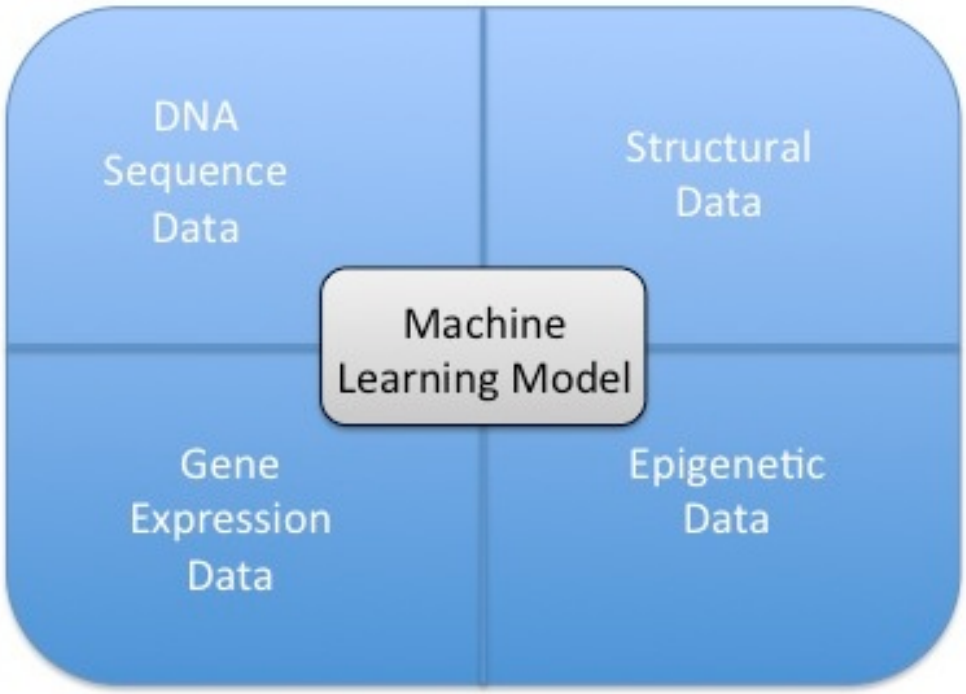

Figure 1- 4 Conceptual representation of combinatorial model. Structural data consisting of nucleosome positioning predictions, Epigenetic data from Ensembl Regulatory Build (Flicek et al. 2010)

The remainder of this thesis presents chapters on the acquisition of the data (chapter 2), the selection of regions to be analysed (chapter 3), the production of the modelling system (chapter 4), the results obtained (chapter 5), and the application of the results to data from GWAS (chapter 6). Additional computational work that contributed to other research projects is outlined in chapters 7. The final chapter 8 summarises the main finding of the thesis, highlights its limitations and proposes areas for further research and application.

2 Data Extraction

In order to develop and test new models for the identification of functional TFBSs using supervised machine learning methods, two types of data are required; (a) dependent variables and (b) independent variables. The dependent variables are those used to label a set of TFBSs as true positives (TPs) or false positives (FPs). The independent variables are those extracted or calculated for each piece of labelled data and are used to model the labelled data.

All the work in this thesis is based on data sets of Human genes and this chapter starts by describing the principal data resource for the human genome, Ensembl (Flicek et al. 2010). Then the dependant variables used to label the data as true and false positives and the independent variables used for modelling are described. In each of these sections the data sources are described generically in the first instance, and then the specific data flows are described in detail. The extraction, calculation and storage of the dependant variables for the labelled dataset is a computationally intensive task. The data has been merged into an analysis database, SETS (Sequence, Expression, Temporal, Structural), which is described in the last section of the chapter.

The modelling predictions are for TFBSs in the human genome and the base data extraction has been principally sourced from Ensembl (Flicek et al. 2010). Ensembl (http://www.ensembl.org) provides a centralised and up-to-date resource for genomes of vertebrate and additional eukaryotic species, including Humans. Data is available interactively via a genome browser, using the menu-driven data-mining tool, BioMart (Smedley et al. 2009), and programmatically via a Perl API (see http://www.ensembl.org/info/docs/index.html). For the

human genome database Ensembl includes functional data from the ENCODE (Encyclopaedia of DNA Elements) project (Becker 2011)

ENCODE is an international research collaboration whose aim is to identify functional elements within the human genome. The pilot phase of the project (that commenced in 2003) targeted 1% of the genome (30 million base pairs) and investigated which experimental techniques could be implemented on the complete genome. The second phase of the project (that commenced in 2007) searched for functional elements in the entire human genome using the key experimental techniques shown in table 2-1 below.

Technique	**Description**
ChIP-Seq	Chromatin Immunoprecipitation combined with massively parallel sequencing to identify protein-DNA interactions.
DNase I Hypersensitivity	Accessible regions of chromatin are sensitive to the enzyme DNase I identifying areas of potential regulation.
DNA Methylation	The addition of a methyl group to nucleotides and the examination of its effect on gene expression.
RNA-Seq	A high-throughput technique sequencing cDNA to determine the RNA content.

Table 2-1 Techniques used by the ENCODE project to discover functional elements of the genome.

The second phase saw the production of 1640 datasets focussed on 24 standard types of experiment for 147 different cell types (Flicek et al. 2012). These results revealed that 80.4% of the genome shows functionality for one or more cell types. The data from the second phase was then made available through the

funcgen database (Flicek et al. 2010) within Ensembl and provides access to experimental data giving information on transcriptional regulation.

To undertake the modelling that forms the basis of this thesis a well-annotated data set of human genes were required. Hence, the genes used were limited to those that had an HAVANA annotation. The HAVANA (Human and Vertebrate Analysis and Annotation) team, within the Wellcome Trust Sanger Institute, aims to create the complete and accurate 'gold standard' annotation for vertebrate genomes including human using in-house computational tools (Wilming et al. 2008). A further requirement for the genes sets used in this thesis was to limit the genes they included to those that were protein-coding genes. As the analysis incorporates data at the transcript level, i.e. gene expression data, every protein-coding transcript for each of the selected genes was used. This resulted in 21,976 genes from the ENSEMBL release GRCh37.3 of the human genome, these genes comprising 98,751 transcripts.

For the data extraction and analysis, the Ensembl Perl API (Flicek et al. 2010) was used. This API (Application Program Interface) gives direct programmable access to the raw Ensembl data allowing extracts by gene/transcript or by specific locations within chromosomes, this is made available via the class hierarchy implemented by Perl's object-orientated system. The core analysis system for this thesis has been coded in Java and it was initially hoped to extract data via JEnsembl (Paterson & Law 2012), the Java API forming part of BioJava (Holland et al. 2008) but at the time of data extraction this function was out of

date and could not be used. However, it should be noted that with the recent release of BioJava 3 (Prlić et al. 2012) that JEnsembl could now be used to extract the data.

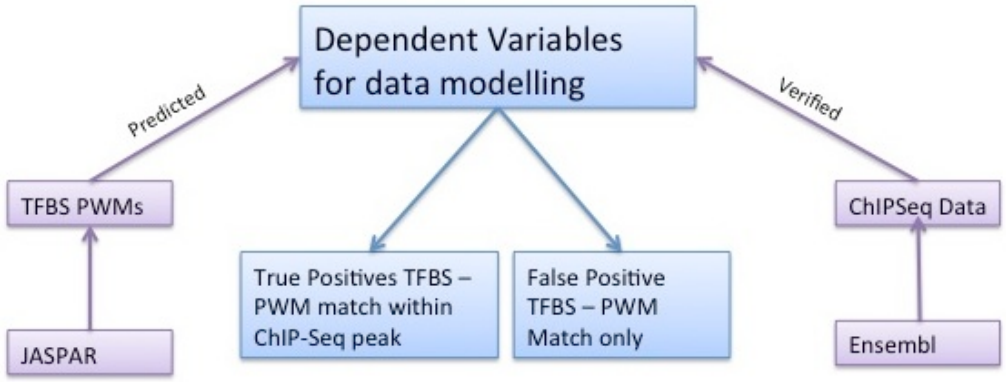

Figure 2- 1 Dependent variables for data modelling. Predicted TFBS obtained from JASPAR are compared to experimentally verified ChIP-Seq peaks from Ensembl to generate true and false positives.

2.1 Dependent Variables: Labeling the Dataset

In order to label a dataset of TFBS as TPs or FPs we require a means of making TFBS predictions computationally and then determining if each predicted site has been observed to be functional experimentally.

2.1.1 Predicted TFBSs

Many TFs show preferences for binding specific sequences of DNA although these TFs can still tolerate a range of variation at different positions (Wasserman & Sandelin 2004). Experimentally derived results, both from functional regulatory gene elements and by randomly examining DNA sequences and determining which are preferentially bound, have been collected and analysed (Jagannathan et al. 2006). A consensus sequence where the most populous bases at each position is taken leads to a loss of data and therefore, the most common method of determining the ability of a transcription factor to bind to a TFBS is via the use of position weight matrices (PWMs).

A PWM is calculated by determining the composition of base pairs at specific genomic positions; these are converted into log-likelihood probabilities calculated thus:

To produce a PWM we firstly need to create a position frequency matrix (PFM) by determining and summing the composition of base pairs at genomic positions, for example:

BP	Pos 1	Pos 2	Pos 3	Pos 4	Pos 5
A	12	3	0	4	0
C	0	0	11	7	0
G	0	9	0	0	0
T	0	0	1	1	12
Total	12	12	12	12	12

Table 2-1 Example Position Frequency Matrix (PFM) showing frequencies of bases by position in sequence.

This data can also be represented by a sequence logo (Crooks et al. 2004)

The following formula (Stormo 2000; Lenhard & Wasserman 2002) is then applied to translate the PFM into a PWM:

$$\text{weight} = \log_2 (((f + \sqrt{N} * p)/(N + \sqrt{N}))/p)$$

Where f = frequency of nucleotide

N = Number of sequences analysed

P = Proportion Expected – 0.25 in case of DNA bases

When applied to the previous example we get the PWM table below:

BP	Pos 1	Pos 2	Pos 3	Pos 4	Pos 5
A	1.735	0	-2.158	0.332	-2.158
C	-2.16	-2.158	1.618	1.025	-2.158
G	-2.16	1.352	-2.158	-2.158	-2.158
T	-2.16	-2.158	-1.051	-1.051	1.735

Table 2-2 Example Position Weight Matrix (PWM) calculated from PFM in Table 2-1.

To produce a similarity measure for a DNA sequence we firstly calculate the minimum and maximum possible scores for a PWM. Scoring the sequence we now want to compare we can calculate a relative score by:

$$\frac{SeqScore - MinScore}{MaxScore - MinScore}$$

to create a ratio.

This calculation is applied both to the forward and the reverse strands, a cut-off for similarity is often taken at 0.75, 0.80 or 0.90 dependent of the stringency of the requirements.

There are two main databases of TFBS predictions using algorithmic methods, (a) JASPAR (Sandelin et al. 2004), and (b) TRANSFAC (Matys et al. 2003).

(a) JASPAR is a non-redundant manually curated collection of PWMs for 23 species released as an open-source product. It underwent a major update in 2010 (Portales-Casamar et al. 2010) and is currently being updated in 2013. At the time of extraction, JASPAR contained PWMs of 490 TFBSs, 76 of which were from Homo sapiens.

(b) TRANSFAC is a redundant collection of PWMs for more than 300 species. It has been a commercial product since 2005, a public version of the data is available but only comprises data captured up to 2005. The public version contains 446 PWMs relating to Humans.

Both the JASPAR and TRANSFAC public databases have been loaded with the raw data consisting of position-frequency matrices (PFM) predictions of consensus sequences of TFBS's for various species, including Homo sapiens. In the final SETS database JASPAR predictions have been used in part due to their being updated more recently but additionally due to the more accurate manual curation.

2.1.2 Experimentally Verified TFBSs

There are currently 3 databases that curate data on TFBSs that have been verified as functional through experimental techniques (a) Funcgen (included within the Ensembl regulatory build) (Flicek et al. 2010) (b) HTPSelex (Jagannathan et al. 2006) and (c) ORegAnno (Montgomery et al. 2006). The introduction of the Funcgen based on data from the high throughput ChIP-Seq technique from the rolled out ENCODE project in 2011, essentially superseded the other two resources which in comparison are of somewhat limited value.

Hence, the Funcgen database will be discussed in detail but HTPselex and ORegAnno will only be outlined, as these have not been used in the final modelling work in this thesis.

a) Funcgen : Ensemble Regulatory build:

Ensembl contains a regulatory build consisting of "best guesses" of regulatory features (Flicek et al. 2010) based on the funcgen database. To construct these datasets, key regions are taken across all cell types to define a set of binding sites thereby marking regions most likely to contain regulatory elements via the ChIP-Seq technique (Jothi et al. 2008; Barski & Zhao 2009). This technique analyses protein interactions with DNA, it comprises chromatin Immunoprecipitation to isolate specific DNA sites in direct contact with transcription factors (and other proteins) with these identified fragments being passed into massively parallel DNA sequencing to identify the binding sites. These regions are limited to 2 kilobases except in the cases of direct overlaps. Specific cell-types are available for some elements but the general nature of the modelling for this project resulted in the MultiCell lines being used. For TFBSs, these are mapped to the publically available JASPAR (Sandelin et al. 2004) PWMs and, using log-odds scores, are compared to random genetic sequences to determine if they are worthy of inclusion into the "best guess" feed. The volume and robustness of this data meant that it was used in the final modelling process (see chapter 4).

b) HTPSelex:

SELEX (Systematic Evolution of Ligands by Exponential Enrichment) is an experimental protocol designed to isolate small populations of bound DNAs from

a random pool of DNA sequences derived by PCR amplification (Tuerk & Gold 1990). It provides a way of finding the in-vitro binding specificities of transcription factors (O + B, 2012). HTPSELEX (Jagannathan et al. 2006) comprises a database of PWMs of TFBSs that have been obtained through either high or low throughput SELEX protocols. As of the December 2012 update, there are data on 12 TFBSs obtained via high throughput SELEX and a further 18 from the original SELEX techniques. This does extend the amount of data we obtained via ORegAnno (Montgomery et al. 2006) (see section c) but is still very limited compared to that now available through the funcgen database within Ensembl. Hence the data was not used in the final modelling.

c) ORegAnno:

ORegAnno is a small but experimentally verified database of actual TFBSs (Griffith et al. 2008). The ORegAnno database includes key fields such as chromosome, start and end position for TFBSs. However, the data it contains (340 TFs that are attached to target genes in the human genome) is again small compared to the funcgen database and has not been updated since February 2008. Furthermore, the website is now unreliable in terms of the accessibility of the data and a third-party application, PAZAR (Portales-Casamar et al. 2009) had to be used to access the data. Hence the data from ORegAnno was not used in the final modelling.

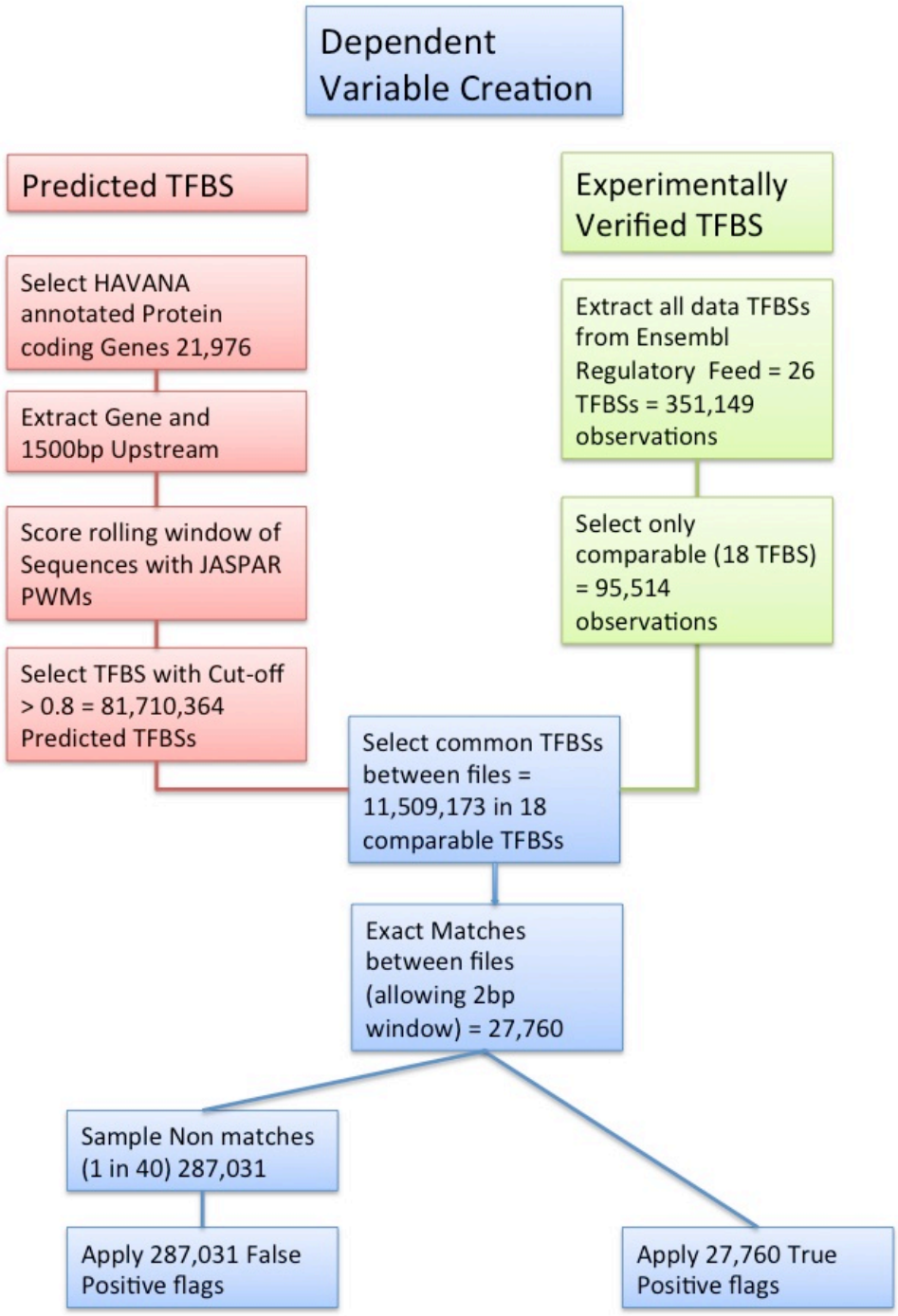

Figure 2- 2 Dependent variable creation. Genetic sequences were scored with JASPAR PWMs, those scoring highest were compared to experimentally verified results from Ensembl to create true and false positives.

2.1.3 A Labeled Dataset of TFBSs in the Human Genome

Taking our 98,751 transcripts and looking at a flanking region of 1500 base pairs upstream of the Transcription Start Site and 200 base pairs into the gene as suggested by the Entropy modelling (See Chapter 3), Perl scripts were produced to query the Ensembl databases, extract the relevant sequences and populate our local database tables. PWMs have then been calculated for flanking regions of all transcripts by taking a rolling window of base pairs the size of the relevant TFBS and calculating a similarity score for each position. This resulted in the scoring of 76 TFBSs * 1700 Rolling Windows * 98,751 Transcripts * 2 strands, or c25.52 billion values. Applying a cut-off similarity score of 0.8 results in 81,710,364 predicted binding sites for the 76 matrices.

On the experimental verified side the total number of TFBSs obtained from the ENCODE project (Ensembl release 66) consists of 351,149 records. When we consider only those JASPAR TFBSs that are consistent between JASPAR and ENCODE, 18 TFBSs, this nets down to 95,514 records. Reducing our predicted TFBSs accordingly we produce net figures of 11,509,713 that can be directly compared. By matching exact positional matches between Ensembl and our PWM based calculated TFBSs we observe 24,313 matches. When we allow a very small 2 base pair window in either direction we can increase this number to 27,760 directly comparable matches. Subtracting these from potential matches we define our dependent variable as true or false positives:

True Positives (ChIP-Seq peak matches with PWMs): 27,760

False Positives (PWMs with no ChIP-Seq matches):

11,509,173 – 27,760 = 11,481,413

For false positives we then used under-sampling on a 1 in 40 basis to address the large imbalance between the two groups resulting in a figure for sampled false positives of 287,031.

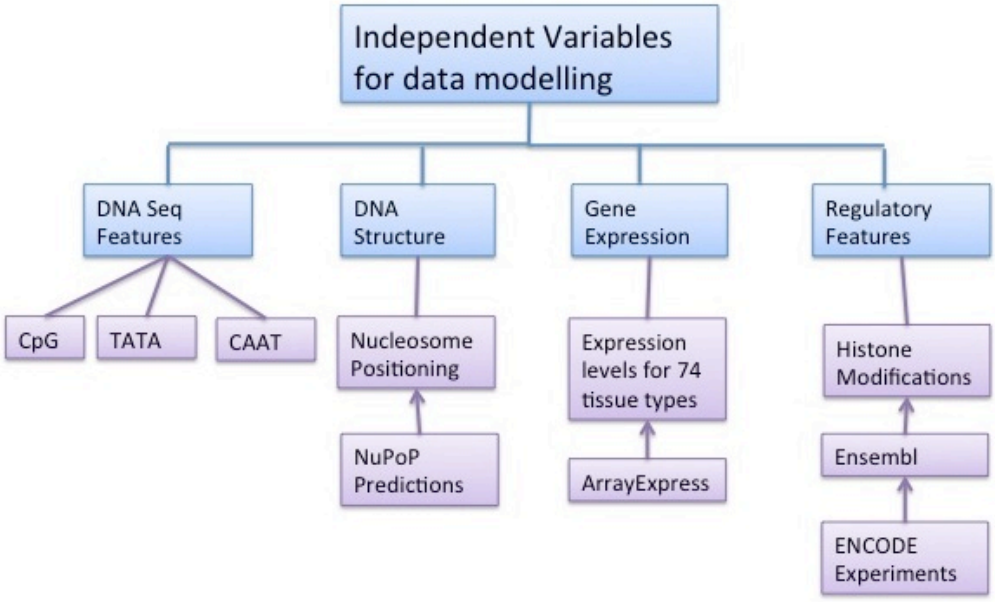

Figure 2- 3 Independent variables for data modelling. Four distinct categories of data have been obtained from varying sources and combined to create the set of independent variables.

2.2 Independent Variables: Data for Modelling

As displayed in Figure 2-3, independent variables were created for each labelled data item based around four related sources: DNA Sequence features, DNA structural features, gene expression and gene regulatory features.

2.2.1 Sequence Data

The content of the 1500 base pairs upstream of the TSS together with 200 base pairs into the gene, as suggested by the entropy modelling, into the gene was analysed for the presence of key elements known to affect the expression of genes.

(a) CpG Islands: CpG Islands affect the ability of DNA to be methylated. The traditional definition (Gardiner-Garden & Frommer 1987) of a CpG Island is a sequence of DNA with a minimum length of 200 base pairs where the % of G and C bases total at least 50% and the observed CG (base C followed by base G) combination is greater than 60% of that expected by chance e.g.

CG/(C x*G) * Sequence Size

Equation 2- 1 Calculation of CpG Islands

The content of the 1500 upstream base pairs was split into ranges of 0-500, 501-1000 and 1001-1500 away from the TSS and the 200 base pairs into the gene were separately examined for CG Content therefore allowing the calculation of whether a potential TFBS was encased in a CpG Island.

(b) TATA-boxes: TATA or Goldberg-Hogness boxes (Lifton et al. 1978) are present in the core promoter of approximately 24% of human genes and enhance the binding of key transcription factors such as TFIID as part of the basal transcription complex, they consist of a consensus sequences of

TATAAA/TTTATA typically within 25 base pairs of the TSS and exposed by nucleosome remodelling (Cairns 2009). This sequence was checked and a binary variable was produced and stored for each potential TFBS on whether a TATA box was present.

(c) CAAT boxes: CAAT boxes are another signal for binding general transcription factors; these patterns are often found 100-150 bases upstream of the TSS. Once more the sequences were analysed and a binary variables produced showing the presence or not of a CAAT box.

2.2.2 Structural Data

A key factor in transcription is the accessibility of genomic DNA to TFs. As described in Chapter 1, segments of DNA are packaged in nucleosomes, complex structures comprising DNA and histone proteins, which can result in DNA sequences being inaccessible to TFs. The DNA sequence itself is predictive of nucleosome occupancy and depletion (Yuan & Liu 2008). Whilst there is a debate over human regulatory sequences such as TFBSs being seen more frequently at areas of high nucleosome occupancy (Tillo et al. 2010) or low nucleosome occupancy (Daenen et al. 2008)), combining these predictions with the other datasets will provide the basis for novel models for the prediction of the functionality of TFBSs.

Two main methods for the prediction of nucleosome occupancy based on DNA sequence information were tested, (a) an executable program from the Segal lab

(Segal et al. 2006) henceforth referred to as the Segal program and (b) a predictive program NuPoP (Xi et al. 2010).

(a) Segal Program: The Segal labs provide an online nucleosome prediction program (Segal et al. 2006). The executable is also downloadable and this was tested. Their models are based around a chicken nucleosome-DNA interaction model and recommend a flanking region of at least 5K base pairs to provide a result.

(b) NuPoP: NuPoP is a software tool produced by Ji-Ping Wang at Northwestern University (Xi et al. 2010). This program is an R/Bioconductor SVM (Support Vector Machine) package that analyses sequences of DNA and produces a nucleosome occupancy scores on a species-specific basis. During testing the R program has been called by Java programs to create results dynamically.

The most effective for modelling purposes was NuPoP as the program is more flexible in terms of sequence lengths, not requiring large flanking sequences that are not scored, and is rapid enough to perform calculations on many thousands of upstream and downstream regions.

All areas of the genome potentially containing a TFBS were scored with the NuPoP algorithm and predictive scores ranging from 0 to 1000, low to high likelihood of nucleosome occupancy were inserted into the database for each potential TFBS.

2.2.3 Gene Expression Data

Summary data regarding levels of gene expression in different tissue types have been sourced from ArrayExpress (Parkinson et al. 2007) at the EBI. The ArrayExpress repository is a publically available database storing results of high throughput genomics experiments. All data generated is MIAME (Minimum Information About Microarray Experiment) (Brazma et al. 2001) compliant, a standard that was created to ensure consistent comparable data is collected across the wide range of microarray experiments. Gene expression studies comprise in excess of 90% of experiments held and Human is the largest represented organism.

ArrayExpress is continually updated but at the time of extraction (Jan 2012) the complete database consisted of 44,775 transcripts with their average expression levels in 80 different human tissue types. These data have been obtained from over 30,000 hybridizations and have been downloaded, merged and stored in local database tables (Parkinson et al. 2007). For modelling purposes, only those tissue types relating to normal adult cells have been included in the dataset hence those tissue types relating to development and disease have been excluded from the analysis. This has resulted in 76 types being available for use.

2.2.4 Regulatory Features

Regulatory data was obtained via two methods, firstly, specific sequences known to preferentially allow epigenetic modifications were examined via the DNA sequences (see 2.2.1) and, secondly the Ensembl Regulatory build was queried to look at modifications observed via experiments.

The complete list of annotated features available on the Ensembl Regulatory Build (Flicek et al. 2010) was examined; wherever experiments had observed features within our proposed regions we were able to create variables.

Most eukaryotic genes use the mechanism of the formation of the RNA Polymerase II (POL II) complex to regulate transcription (Schones et al. 2008). POL II forms part of the pre-initiation complex (PIC) that recognises and attracts key basal transcription factors to initiate mRNA transcription. CTCF plays a key role in transcription regulation predominately by acting as a repressor although it also has a role in regulating the 3D structure of chromatin (Bickmore 2013).

Histone modifications also play key roles in regulation with methylation of histone tails typically inhibiting transcription whilst demethylation gives access to the transcriptional machinery (Tost 2009).

To apply this data to the SETS database, the modifications had to be relevant to all cell types. To this end the Multicell features were used from the Ensembl Regulatory Build. This processes data from ENCODE experiments and computes

MultiCell features which can be used independently of cell types (Flicek et al. 2012). The following epigenetic modifications were selected as they had the highest representation in the dataset, shown in table 2-2 below together with their counts from the Regulatory Build:

Regulatory Feature	**Description**	**Modification**	**Count**
DNAse1	DNAse1 Hypersensitivity		1,255,253
PolII	RNA Polymerase II transcription factor		248,664
CTCF	Transcription Repressor		652,081
H3k4me1	Histone 3 modification	activation	120,671
H3k4me2	Histone 3 modification	activation	267,463
H3k4me3	Histone 3 modification	activation	365,163
H3k9ac	Histone 3 modification	activation	183,917
H3k27ac	Histone 3 modification	activation	240,528
H3k27me3	Histone 3 modification	repression	54,397
H3k36me3	Histone 3 modification	elongation	749,713
H4k20me1	Histone 4 modification	repression	28,977

Table 2- 1 Epigenetic Features extracted from the Ensembl Regulatory Build.

The values obtained for these variables are raw signal values with troublesome peaks, those for example where values are less than control, pre-edited out by avoiding ENCODE identified problem areas (Becker 2011), see figure 2-4.

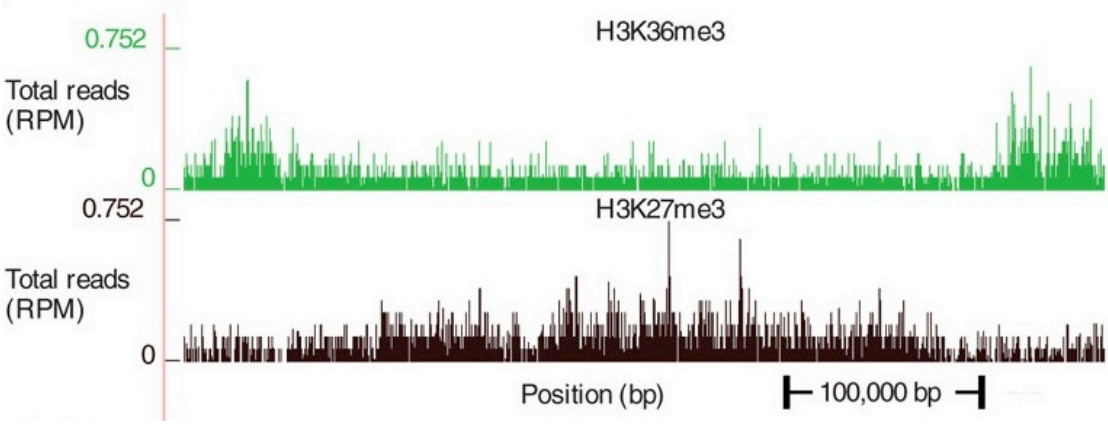

Figure 2-4 ChIP-Seq peaks example (Figure adapted from Nature Methods: Computation of ChIP-Seq peak types from various experiments, Pepke (2009).

2.2.5 A Dataset of Independent Variables

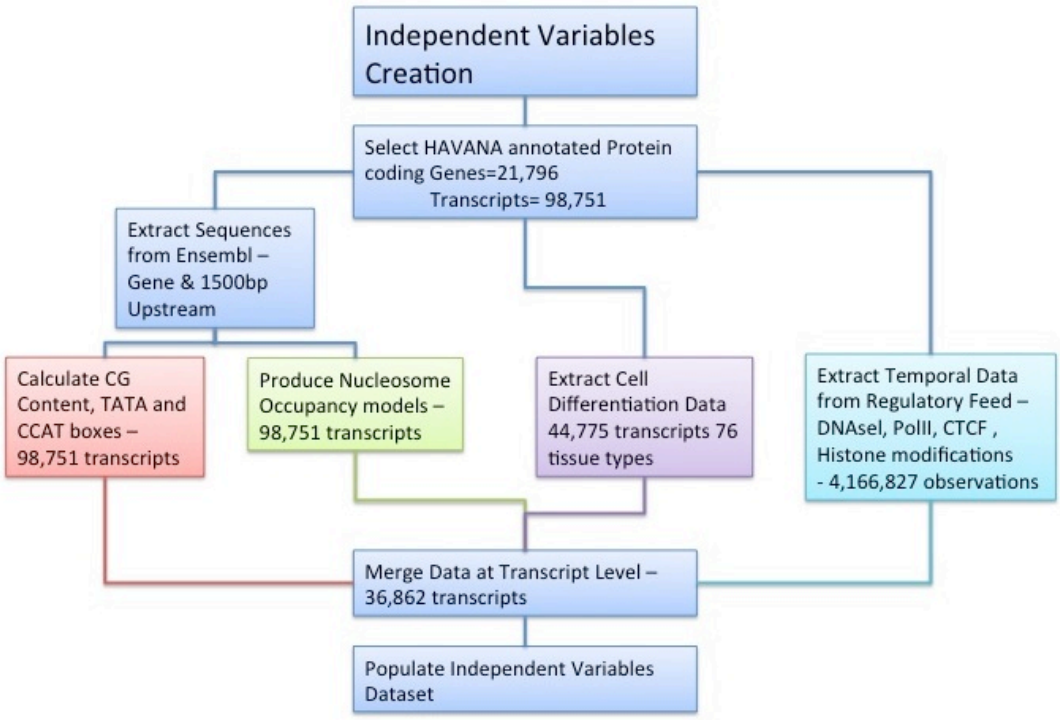

Figure 2- 5 independent variables for data modelling summary.

Representation of the extraction of data from key sources.

The data from the above four areas has been merged to provide the independent variables to go forward to the modelling phase. For the 98,751 HAVANA annotated transcripts we can produce full data from the Ensembl based sequence and structural sides but are limited to those transcripts that have expression data on the ArrayExpress database. Although there is gene expression data relating to 44,775 human transcripts this reduces to 36,862 when we match to the main 98,751 transcripts. This is due to the previous applied HAVANA annotation and protein-coding restrictions that have been

applied to the main selection. The final stage is then to merge all temporal information that matches these transcripts.

2.3 SETS (Sequence, Expression, Temporal, Structural) Database

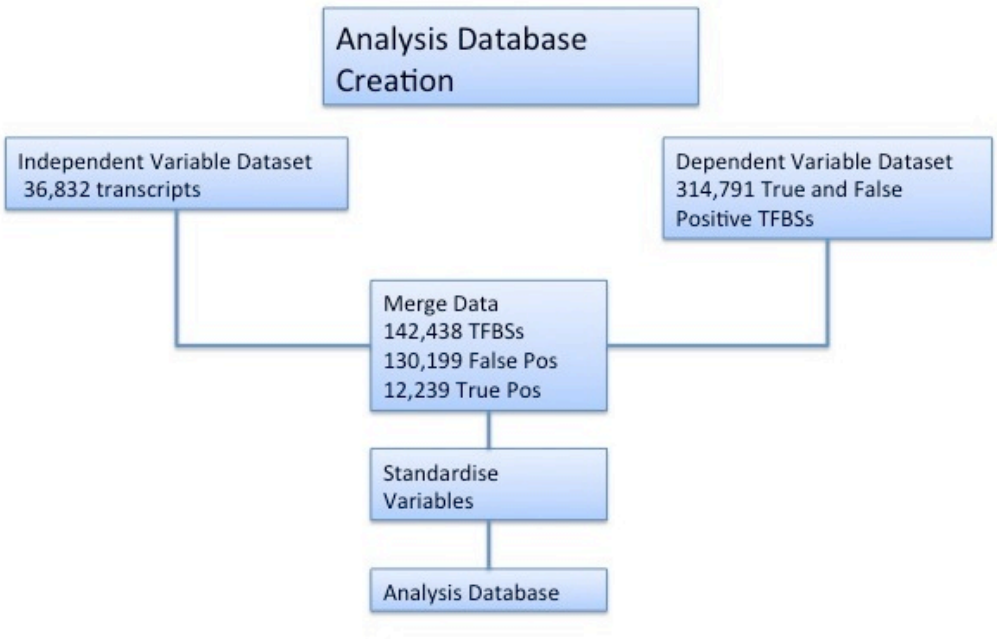

Figure 2- 6 SETS Database Creation. Combination of dependent and independent variables.

As the modelling would require a complete set of variables for each observation, true and false positives could only be used where we have complete information relating to their transcript. When data from our four areas of independent variables where combined, a complete record was obtained for 36,822 transcripts limited predominately by the availability of gene expression data. As

shown in Figure 2-6, the resultant merging process resulted in an analyzable dataset with 142,438 observations of which 130,199 were false positives and 12,239 were true positives.

Prior to modelling all data has been standardised and the resultant z-scores stored in a specific MySQL table to facilitate the modelling using the standard formula below:

$$z = \frac{(x - \mu)}{\sigma}$$

Where μ = mean and σ = standard deviation

Equation 2- 2 Z-score calculation

This normalisation is standard procedure in neural network modelling. It is most appropriate where data is Gaussian, as is the case for the structural and sequence data. The Java language has been used to create access methods, utilities and reports. This language has been chosen due to its speed of processing large datasets and its inherently object orientated nature. Custom Java classes have been written to allow access to data either locally, over the Sussex University network, or using secure remote Internet access via HTTP. For specific access to key datasets, Perl modules and scripts have been built to access their relevant APIs. These factors, when combined, allow the creation of a modular and flexible system. A code and class summary is presented in the appendix.

The database has been assembled in the MySQL (http:/www.mysql.com) open source database management system (DBMS). MySQL is a fully functional open-source DBMS providing standard SQL access to database tables using virtually any programming language including the Java and Perl languages used for this project.

2.4 Data Summary

The extraction of independent variables, as summarised in figure 2-4, resulted in 93 variables for each labelled TP and FP TFBS. Table 2-3 provides a summary of these independent variables that have been used for modelling as described in Chapter 4.

Summary of Data Fields used in Modelling

Data Field(s)	Source	Description	Storage Format
TranscriptID	Ensembl		AlphaNumeric
Positioning	Ensembl	Chromosome, Offset start and End, Strand	Numeric
TFBSID	Jaspar	TFBS identified by PWM	Alphanumeric
EXPVER	Ensembl Regulatory	Has the TFBS been experimentally verified?	Binary
CpG Islands	Ensembl	CG content, CpG islands in DNA regions	Binary – Is TFBS in CpG Island
CAAT Boxes	Ensembl	CAAT box in promoter regions	Binary
TATA Boxes	Ensembl	TATA box in promoter regions	Binary
Nucleosome Positioning	NuPop Calculated	Likelihood of Nucleosome at TFBS position	0-1000 – result of NuPoP Prediction
Entropy	Calculated	Entropy Score of 256 base pairs centred around TFBS	0-1 – standardised entropy score
DNase1	Ensembl Regulatory	DNase Hypersensitivity, experimentally observed	ChIP-Seq signal strength 0-28
POLII	Ensembl Regulatory	RNA Polymerase II transcription factor, experimentally observed	ChIP-Seq signal strength 0-28
CTCF	Ensembl Regulatory	Transcriptional repressor, experimentally observed	ChIP-Seq signal strength 0-28
Histone Modifications	Ensembl Regulatory	Various Histone modifications experimentally observed	ChIP-Seq signal strength 0-28
Expression Levels	ArrayExpress	Average expression levels in 74 different tissue types	Numeric - Standardised average expression levels

Table 2- 2 Summary of Data held on SETS database.

2.4.1 Additional Data

Other data resources were used both for the linking of disparate datasets and for initial reports and testing.

a) GO Slim: The cut down version of Gene Ontology (Ashburner et al. 2000) terms have also been used during the modelling testing phase. This dataset allowed the comparison of reports and models against genes and transcripts with different attributes within their controlled vocabulary of terms.

b) David (Huang et al. 2009): This bioinformatic resource provided by the NIH (National Institute of Health) provided the David ID that allowed various tables to be linked even when indexed by different identifiers, for example:

- Ensembl Gene ID
- Affy ID for microarray experiments
- ENTREZ Gene ID
- REFSEQ

2.5 Summary

This chapter has detailed how in excess of 6GB of publically available data from the Human genome has been collated and processed to create a relational database. The dependent variables have been used to label predicted TFBSs as TPs or FPs. Independent data has then been collated for each labeled TFBS and includes information based on DNA sequence features, DNA structural features,

gene expression, and regulatory features. This data has been stored in a relational database enabling the data to be used for the machine learning modelling phase of the current work described in Chapter 4.

3 Entropy

When considering which areas of the genome should be examined for potentially functional TFBS, an initial question was how many base pairs upstream and downstream of the transcription start site (TSS) of a gene should be searched. Many studies have looked at a variety of different sequences lengths that typically have focussed on an area within 1000 base pairs of the TSS (Hannenhalli 2008; Veerla et al. 2010). In addition, the initial ENCODE project found that many regulatory elements can be found in a symmetric pattern around the TSS (Birney et al. 2007) therefore an initial investigation was performed to examine the most appropriate size of these regions.

Within the human genome it has been estimated that 5% of DNA is under selection pressure (Waterston et al. 2002), but only 1.5% is estimated to be coding (Lander 2011). Genomic variation or information content (IC) has been used in various studies to estimate selection pressure on DNA sequences in various species, for example, intergenic DNA in H. *sapiens* (Mu et al. 2011), and introns in D. *melanogaster* (Haddrill et al. 2005)and C. *elegans* (Prachumwat et al. 2004). The information content of a sequence can be calculated using entropy formulae adapted from information theory (Schneider 2010). Although different conclusions have been reached by various studies (see Table 3-1), entropy, as a measure of complexity, has been used to suggest the areas of the genome that should be concentrated on when looking for functional TFBSs.

Shannon (Shannon 1949) introduced the concept of entropy into information theory in 1949, this determined the limits of lossless compression, or how much information data contains. Topological entropy is a variant of Shannon entropy

that looks at the number of observed against the number of possible sequences of data (Adler 1979). Applying this entropy definition to a rolling window of DNA gives a single value per sequence thus making this method suitable for applying to the dataset in the current work. This was applied to show that coding DNA had lower entropy than noncoding DNA (Koslicki 2011).

This chapter details the process of applying entropy measurements to the human genome and discusses the application of results to the selection of regions to be searched for functional TFBSs. Specifically this chapter looks at whether entropy can be used as a measure to (a) differentiate exons, introns, and intergenic DNA, (b) examine variation within gene promoters comparing known functional regions and unclassified sequences of DNA, (c) assess the effect of genomic indicators such as GC content, nucleosome occupancy, and presence of TATA boxes, (d) examine the difference between different types of genes, housekeeping and tissue specific, and (e) differentiate functional vs. non-functional TFBSs.

Study	Entropy Calculation	Dataset	Conclusion
(Colosimo & De Luca 2000)	Linguistic complexity	16 DNA sequences including eukaryotes (5 human) and prokaryotes (< 2650bp in length)	native DNA < random DNA
(Troyanskaya et al. 2002)	Linguistic complexity	21 prokaryotic genomes	C > NC
(Liu et al. 2008)	Lossless compression	Human genome	C > NC
(Karamanos et al. 2006)	Topological	2 viral genomes and 4 human gene regions (max ~73K bp)	C > NC
(Koslicki 2011)	Topological	Human genome, 100 longest intron and exon sequences from 23 chromosomes	C < NC (I)
(Mantegna et al. 1995)	Shannon	2 phage genomes, 2 viral genomes C.elegans Chr III: Yeast Chr III & XI 6 E.coli, 3 mouse & 9 human sequences	C > NC
(Stanley et al. 1999)	Shannon	4 Yeast Chr III, VI, IX, XI Primates in GenBank	C > NC (I) (yeast) C =NC (primates)
(Mazaheri et al. 2010)	Shannon	C.difficile (G+C 29.1%) genome B.bacteriovorus (G+C 50.6%) genome	C < NC (IG)

Table 3-1 - Comparison of previous studies applying entropy definitions to estimate the information content of DNA sequences. Five studies conclude that coding DNA (C) has greater IC then noncoding DNA (NC). (I = intronic, IG = intergenic)

3.1 Dataset Extraction

Genes were extracted from Ensembl human genome assembly GRCh37.6 using their application programming interface (API). These genes were firstly limited to those with an HAVANA (see http://www.sanger.ac.uk/) annotation and secondly to those that did not have upstream regions that overlap with other genes. The definition of non-overlapping being that at least 30,000 base pairs of separation between the TSS of one gene and the 3' UTR of the proceeding gene. This dataset comprised 12,259 genes and was designated HAV_12259.

Later chapters of the thesis are focussed on the detail of selecting data from publically available data sources to create variables to be used to annotate DNA sequences. A subset of those techniques (described more fully in Chapter 3) were utilised to add variables to the HAV_12259 dataset to allow analysis in several areas.

A subset of the HAV_12259 dataset were classified into housekeeping (HK) (also known as constitutive) genes and tissue specific (TS) (also known as facultative) genes. HK genes are predominantly involved in basic cell functions and TS genes relate to specific functions in distinct cell types. These subsets were based on a meta-analysis study of 104 microarray datasets (Chang et al. 2011) looking at 1,431 samples from 43 different human cell types that identified 2064 HK and 2293 TS genes. The numbers of genes that matched to the HAV_12259 dataset were 507 housekeeping and 596 classified as tissue-specific.

For all of the genes in the HAV_12259 dataset, sequences were extracted for all exons and introns and also the upstream and downstream intergenic regions via the Ensembl API.

In addition to allowing the calculation for Topological Entropy for varying subsequence sizes, these extracts permitted the calculation of CG content for promoter regions. This was achieved by examining a rolling window of 200 base pairs and calculating the percentage of C or G bases within.

3.2 Calculating Topological Entropy

The process of calculating Topological entropy (H_{top}) involves the computation of a complexity function to determine how random the sequence is, or how difficult it is to compress, the sequence is. The observation of a large number of different subsequences results in a higher entropy value as this would have high information content and be more difficult to compress. Conversely, a small number of different subsequences would be easier to compress, have smaller information content and result in a smaller entropy value.

For the DNA alphabet of four bases {A,C,G,T}, a sample of DNA can contain 4^n possible distinct sequences where n is the length of the sequence. To calculate H_{top} a minimum number of sequences must be examined to ensure each distinct sequence could be observed, this is achieved by selecting a length of DNA of $4^n + n - 1$ bases and looking at a rolling window of n bases within that sequence. For example, to calculate H_{top} for a five base pair subsequence, a sequence of 1028 base pairs is examined via a five base pair rolling window allowing 1024 unique

sequences. The final step of the calculation is to take the $\log_4$ (number of DNA bases) of the number of different observed subsequences (OS) and divide by the number of bases in the rolling window (n). The result of the observed calculation can then be compared to an expected value for the number of bases in the rolling window.

Observed Topological Entropy

$$O[H_{top}] = \frac{log^4(OS)}{n}$$

Expected Topological Entropy

$$E[H_{top}] = \frac{log_4(4^n - 4^n(1 - 1/4^n)4^n)}{n}$$

Equation 3- 1 Calculation of Observed and Expected Topological Entropy

To check the accuracy of these equations, results were compared to those obtained using the Mathematica code provided as supplementary data to the Koslicki paper (Koslicki 2011).

3.2.1 Transcription Factor Binding Sites (TFBSs)

Details of TFBSs were added to additionally describe the HAV_12259 dataset. Position Weight Matrices (PWMs) from JASPAR (Portales-Casamar et al. 2010) were used to score 1500 base pairs upstream and 200 base pairs downstream of the TSS of the 12,259 genes and a cut-off similarity score of 0.80 was applied. JASPAR PWMs were selected due to their being used by the ENCODE project and

also the more recently updated data provided on their database. Those PWMs that occurred within a ChIP-Seq peak as sourced from the ENCODE project (Becker 2011) were designated a true positive (TP) whilst those not within these peaks were designated as a false positive (FP). This process was necessarily limited to those TFBSs analysed by the ENCODE project and available as a JASPAR PWM, this resulted in 18 TFBSs being analysed.

3.3 Measuring Entropy in the Human Genome

As infinite sequences of DNA are not available to analyse, finite Sample effects can influence results (Koslicki 2011), therefore entropy was calculated based on different sequence lengths permitting different subsequences to be analysed in appropriate rolling windows.

3.3.1 Range of Sequences Analysed

Based on the required sequence length from the equation $4^n + n - 1$, sequences of 259 to 16,391 base pairs were analysed, given n values between 4 and 6 (see table 2-2). Exons in the HAV_12259 dataset mainly consisted of sizes with n values between 4 and 6, whilst Introns, although permitting larger values to be analysed, were examined in the same range. The analysed intergenic DNA comprised a minimum of 30,000 base pairs as defined in section 2.1. Random selections were made from the intergenic DNA to create sequences between 259 bp and 16,391 bp so they could be compared to the introns and exons in the n = 4 to n = 6 range. The java.util.Random class was used as a random size generator and resulted in the creation of 63,771 sequences.

3.4 Results

Relative entropy of sequences was compared for subsequence size and type of sequence, type of gene, and TFBSs in terms of true positives and false positives. The amount of CG content has also been looked at for the different types of genes examined.

3.4.1 Entropy Comparison by Sequence Size

As clearly seen in figure 3-1, mean entropies increase as the analysed sequence size increases. This is a known issue as the calculations were originally intended for infinite sequence lengths. It is therefore important that all results should be compared for same size sequences. Restricting results to those where at least 2000 introns and exons were available (see table 2-2), 78.9% of exons have significantly higher entropy than introns ($p<2.2\ e^{-16}$ observed via t-test comparison of means) and hence considerably higher information content. This was particularly the case in sequences lengths of n=4 and n=6 although comparisons where n=5 were not statistically different. Comparisons of intergenic mean entropies proved inconclusive with relationships between intergenic, exons and introns varying for different sizes of sequences.

Sequence Lengths $4^n + (n-1)$	N	Exons	Introns	Entropy E	Entropy I	Entropy IG	Summary I/E	Summary IG/E/I
259	4	104476	73224	0.887	0.885	0.887	E>I	(IG=E)>I
1028	5	28615	102560	0.906	0.906	0.906	E=I	(I=E)>IG
4101	6	2412	65116	0.919	0.918	0.918	E>I	(IG=I)<E

Table 3-2 Mean topological entropy values for exons (E), introns (I), and intergenic (IG) sequences of N base pairs. Calculations performed on categories > 2,000 introns and exons

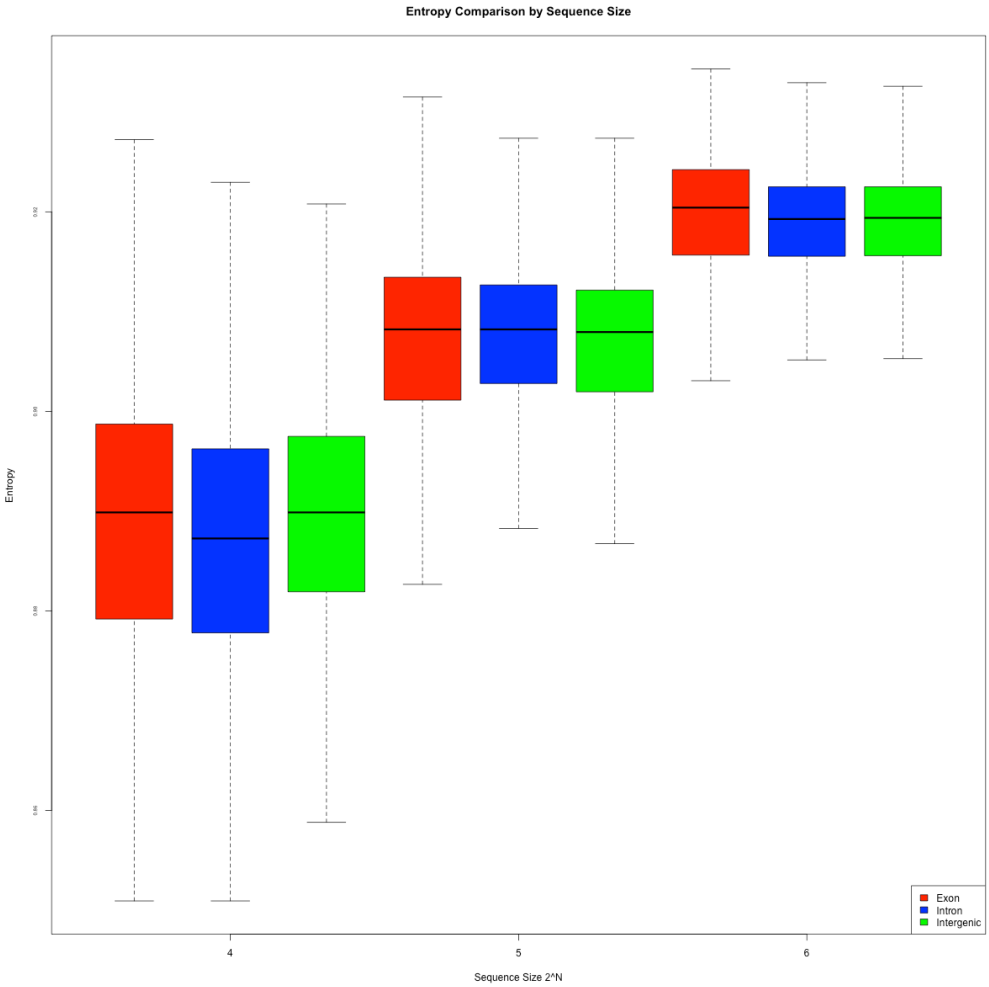

Figure 3-1 Mean topological entropy distributions for exons, introns and intergenic across the human genome for 3 sequence length categories

3.4.2 Entropy Comparison by TFBSs

Comparisons of Transcription factor binding sites were analysed by comparing those genes where True Positives were observed, those where False Positives were observed and all genes as shown in Figure 3-2 (genes were potentially counted multiple times if TPs and FPs were observed for the same genes). As can be seen, a large dip is observed from approximately 1200 base pairs upstream to approximately 300 base pairs into the gene. To a certain degree this is a factor of the definition of true and false positives, in both cases, a PWM from

JASPAR (Portales-Casamar et al. 2010) has been observed, the TP positives having had this PWM prediction verified by ENCODE (Becker 2011) data. This will naturally reduce the number of unique sequences seen within the DNA sequence and hence reduce the entropy, however this does support the examination of the range 1500 base pairs upstream to 200 base pairs in the gene that has been used for the main analysis.

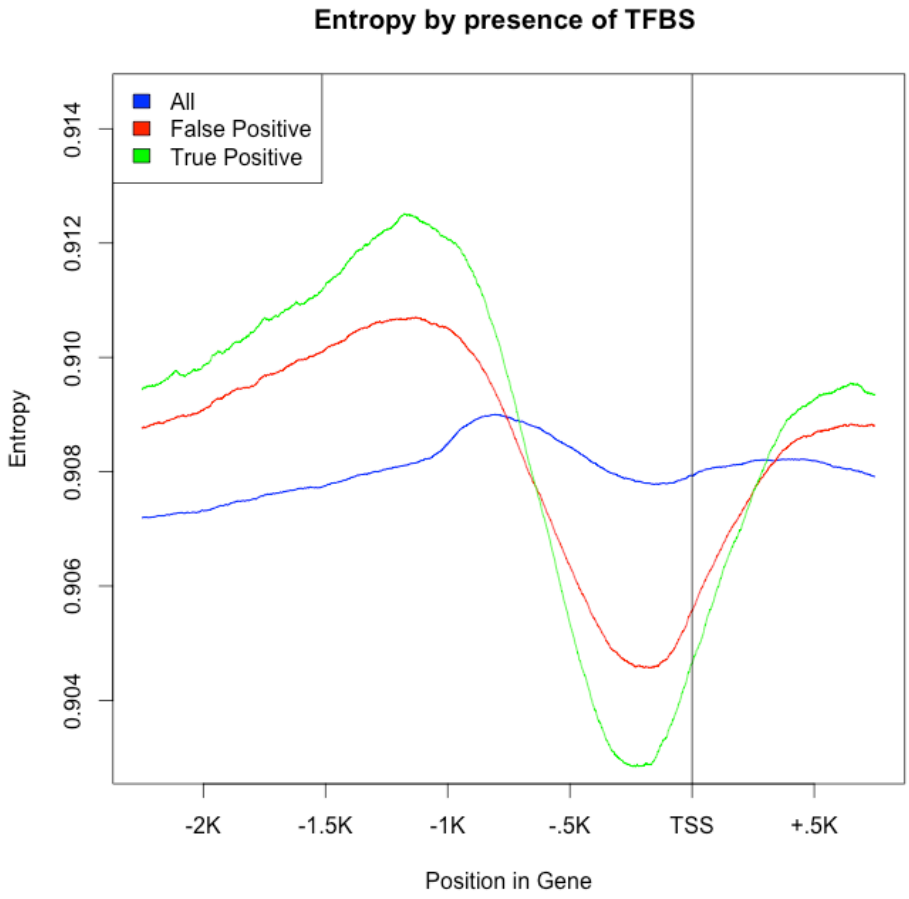

Figure 3-2 Mean topological entropy distributions for all genes, those with a FP TFBS, and those with TP TFBS.

3.4.3 Entropy Comparison by Gene Classification

The entropy profile of the complete HAV_12259 dataset shows two interesting areas. A minima region of approximately 100bp centered near to 225bp upstream of the TSS and a maxima region of approximately 400bp centered 800 bp upstream. Comparing figures 3-3 and 3-4 we can see that this is not completely explained by the variations in GC content observed as the TSS is approached. As also seen in figure 3-3, the tissue specific genes have a similar profile, if flatter, than that of all genes, however the housekeeping genes show a major increase in entropy within the 2k base pair upstream to the TSS region. When compared to the TS genes in this region the HK have significantly higher entropy with a p-value of $p < 2.2^{e-16}$.

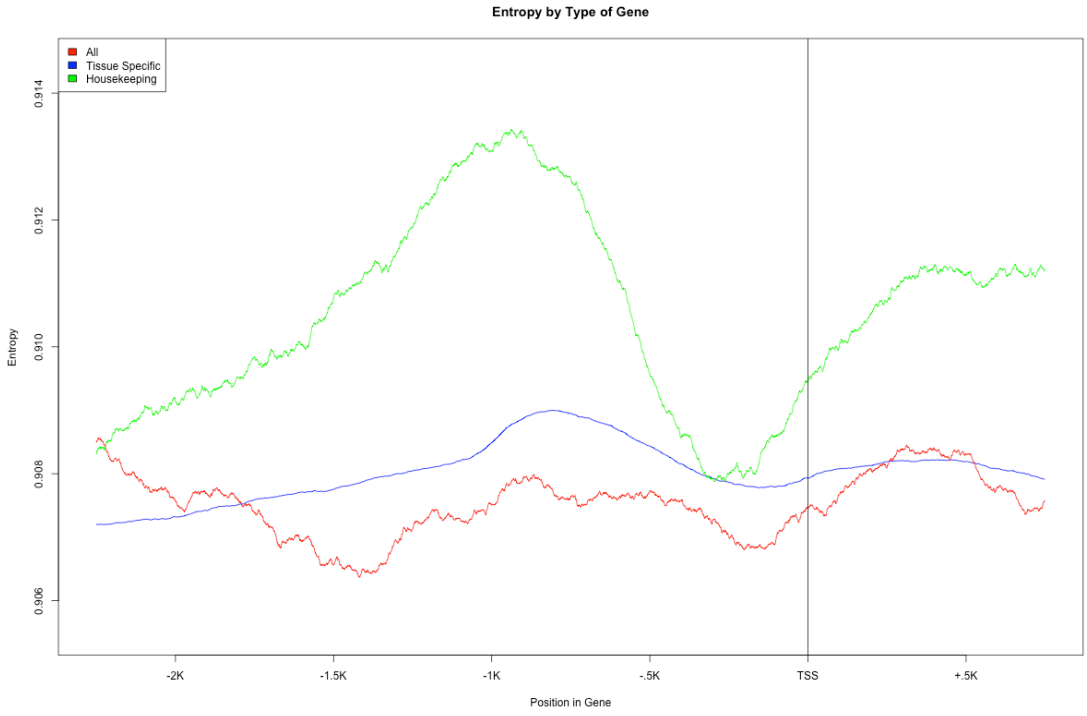

Figure 3-3 Mean topological entropy by all genes, housekeeping and tissue-specific genes.

3.4.4 CG content by type of Gene

Although overall the human genome has a lack of CG dinucleotides to that expected proportionally, many promoter regions contain higher than the background level potentially allowing epigenetic methylation modifications (Saxonov et al. 2006). GC content was therefore examined in terms of proximity to the TSS and separate profiles produced for tissue specific and housekeeping genes. As seen in figure 3-4, a general increase in the percentage of G and C bases can been as the upstream region approaches the gene; this then starts to tail off after approximately 200 bp into the gene. This general profile appears to hold true for both of the subsets of tissue-specific and housekeeping genes.

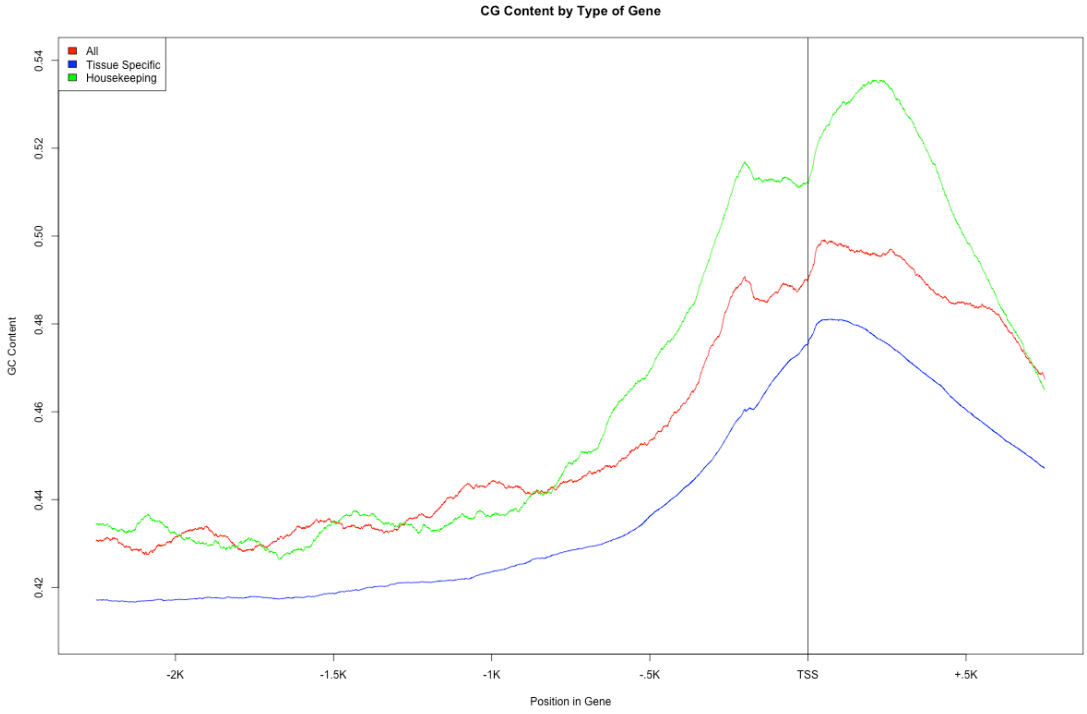

Figure 3-4 % of C and G base pairs observed by all genes, housekeeping and tissue-specific genes.

3.5 Discussion

Identifying functional elements within noncoding DNA is a complex problem and entropy calculations have been previously applied with varying results, see Table 3-1. Large scale projects, especially ENCODE (Becker 2011), have also questioned the amount of "junk" DNA and reported that 80% of the human genome could be assigned a function, although this is still being debated (Graur et al. 2013), (Niu & Jiang 2013). This chapter has looked at a systematic approach to examining different elements of the genome such as exons, introns and intergenic DNA alongside factors such as the presence of TFBSs, CG content, and different types of genes.

Genetic regulation via promoter regions is complex. Combinations of nucleosome occupancy, epigenetic factors such as histone marks, CpG islands and specific known elements such as TATA boxes play a part, in addition to the DNA sequence. Hence results have not shown a straightforward link between entropy and function. There are several other issues that need to be considered when it comes to measuring the entropy of a DNA sequence. For example, if a base was chemically more likely to mutate to another (a C being statistically more likely to mutate to a G for example), this would reduce the number of unique sequences and hence decrease the entropy of a sequence. A further issue that could influence the results is the number of repeating sequences in the human genome, this has been estimated at 69% (de Koning et al. 2011) and again has the effect of reducing the entropy of the analysed DNA.

The current results such as 78.9% of exons having significantly higher entropy and information content than corresponding Introns and differences between classes of genes shows the utility of this measure within promoter regions. Furthermore, the profiles shown in figures 3-2 and 3-3 give confidence that the areas analysed for the presence of true and false positive TFBSs (1500 base pairs upstream and 200bp into the gene) are reasonable choices for the parts of the genome to be analysed in the remainder of the thesis.

This work forms the basis of a paper submitted to Genome Research (Jan 2014).

Chapter 4 details the processes concerned with the setting up of the machine learning modelling environment.

4 TFBS Modelling Setup

The extraction, transformation and data loading (ETL) for the dependent and independent variables have been described in Chapter 3. The next step was the development of the modelling environment. Although software for machine learning techniques was available, specifically in terms of a variety of R packages and workbench based software such as WEKA (Hall et al. 2009), the open source data mining software package, it was decided that for complete flexibility the modelling environment should be created in the Java language. Additionally, results obtained from machine learning software tend to be presented as a "black box" solution whilst producing a customised environment allowed for increased understanding of how the results were generated. A framework was used to both inform the structure of the required classes and to test initial results. The framework that was chosen was the Encog Machine Learning Framework v 2.0 (Heaton 2010). This chapter examines the production of code for the analytical system; it looks at the use of the Encog framework, the classes produced, the modelling techniques available and the completion of the modelling environment.

4.1 Production of the Analysis Environment

The chosen programming environment was Java v1.6 (see http://www.oracle.com/technetwork/java/index.html) running inside the eclipse IDE for Java Developers v1.2.1 (Shavor, Sherry 2003). Java was chosen due to the combination of speed of coding against speed of execution and also because of familiarity with the language.

The Encog machine learning framework is produced by Heaton Research and made available as Free Open Source Software (FOSS) under the Apache license. The original Encog libraries were based around the classes used in the book "Introduction to Neural Network with Java" (Heaton 2010), which were turned into an open-source project resulting in various contributions from different developers. This framework now supports many algorithms including Support Vector Machines (SVM), Hidden Markov Models (HMM) and Bayesian Networks. At the time of the production of the modelling environment however, in the latter months of 2011, the framework was based around Clustering (for unsupervised training), and Neural Networks, together with associated training methods such as Genetic Algorithms and Simulated Annealing for supervised training.

The main reasons for the use of the Encog framework were (a) the classes informed the design of the analysis system, (b) various code and utility classes could be used having already been tested and proven to produce efficient models and (c) early results could be tested against those obtained using the Encog library classes. In general, classes pertaining to the structure of the networks were exploited along with utility code to carry out functions such as checking activation functions. The classes concerned with the Encog workbench, input, output, normalisation and reporting of data were not used. These classes were combined with various custom written ones to handle the specific data and modelling requirements of the system.

4.1.1 Class Structure

The classes of the system can be broadly split into three categories, (a) the group handling control, data manipulation and reporting, (b) those required for the model structure and (c) classes that implement the training technique. A fourth set of classes and methods are shown as the Genetic Algorithm technique requires a different scoring system and requires a distinct approach. Utility classes are detailed as a fifth group. The key classes are detailed below; key methods are discussed excluding their utility methods and those concerned with getting and setting variables

a) Control Classes

The class Proj1 controls the modelling process as shown in Figure 4-1. It firstly extracts the data from the database via the DBExec method of the DB Connect class and creates two dimensional arrays of both the pre-normalised independent data, and a second array containing the dependent variable. Proj1 then creates the feedforward network structure for the modelling and starts the training processes that are detailed below. This class also has responsibility for logging the model progress and then storing a serialised or compressed model that can be verified. The ProjValidator class queries the database to create a new random set of cases and then inflates the serialised model to apply the weights to this data. Comparisons can then be made to determine the accuracy of the model.

Proj1 also has responsibility for managing batch processing. The process of creating an individual model typically took a number of hours to complete due to the large number of iterations normally required to make a useful model. The number of combinations of parameters that needed to be adjusted was also large, therefore utilising overnight and weekend runs was paramount and batch processing was built into the controlling classes.

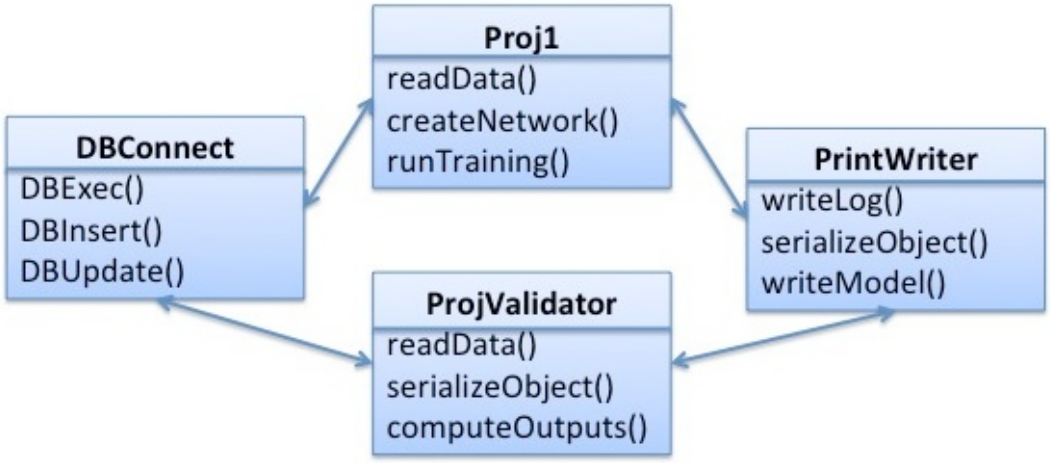

Figure 4-1 Partial UML (Unified Modelling Language) diagram of classes used in controlling machine learning models and their main methods

b) Model Structure Classes

The FeedForwardNetwork class is responsible for creating and managing the various layers that make up the model. Whilst there will always be an input layer to hold the initial variables and an output layer to hold the current state of predictions, there can potentially be several inner or hidden layers. The hidden layers consist of a variable number of "neurons" or hidden variables that receive

inputs from the layers preceding them and either fire or not to provide weights to the final prediction.

Although several hidden layers were permitted by the classes, in practice only one was used to, in part, prevent a common pitfall of neural network modelling, that of over fitting (Tetko et al. 1995). In addition to having too many layers, if too many input variables or neurons within the layers are used, the model is in danger of being just a categorisation tool and therefore will perform very badly when it comes to validation. Over-fitting occurs when more variables are added to a polynomial curve fitting process. Increasing the complexity of the equation can reduce errors but the results will not generalise well. An immediate level of complexity will provide the best results with Feedforward Neural Networks being less susceptible to this issue (Bishop 1995).

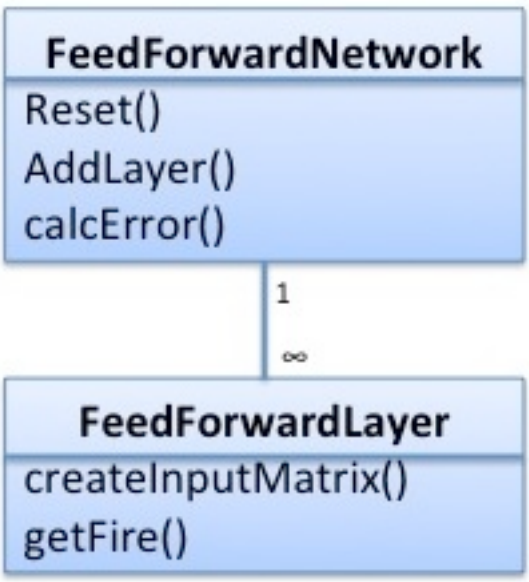

Figure 4-2 Partial UML (Unified Modelling Language) diagram of structural Classes for holding the network during calculation iterations and their main methods

c) Training Classes

These classes control the actual scoring and production of the models. The Encog framework provided six training techniques that used various types of backpropagation networks (see section 4.2 for further explanation). For the six techniques used (Figure 4-3), the functionality is provided by implementations of the Train interface. After the initial random weights were applied, the deltas (the differences between observed and expected values) were calculated based on the relevant activation function. Synapses were created to link the layers and initial random weights were applied to perceptrons within these layers, the deltas (the differences between observed and expected values) were calculated based on the relevant activation function and subsequently amended with each iteration of the model. Based on the aggregated values of these weights on the input variables, neurons can either fire or not fire thereby scoring the next layer. They were then amended by percentage amounts that were taken from input parameters to the model.

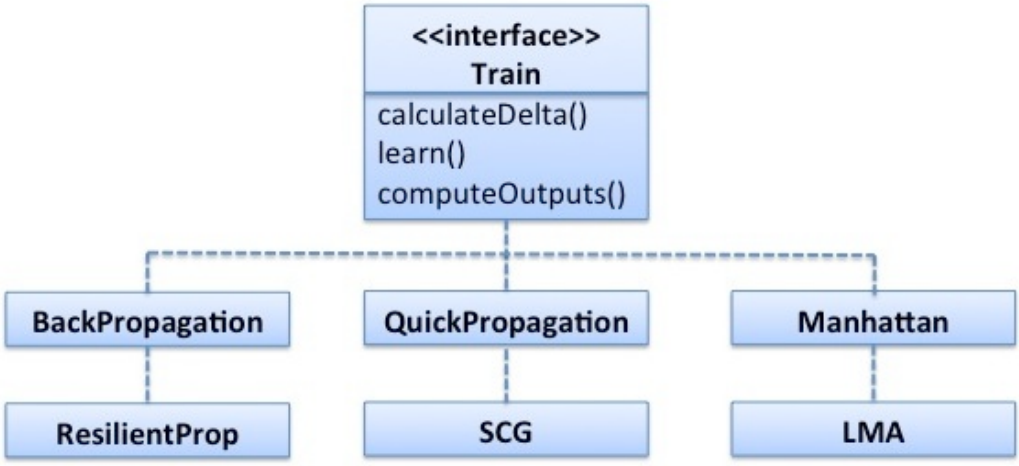

Figure 4-3 Partial UML (Unified Modelling Language) diagram of classes required for training the model with their key methods

d) Genetic Algorithm Classes

In addition to the above methods, the Genetic algorithm method of amending weights was tested. This was performed via a different approach and therefore required a different set of specific classes. Genetic algorithms simulate a single chromosome with variables acting as "genes". The initial population comprises the training set supplied with a random set of weights for each variable/gene. The fitness of individuals in the population can then be calculated by their accuracy in predicting the output.

A percentage of the population can be selected to mate based on their ability to predict the output variable, for example we may select the best 25%. This breeding population undergoes a simulation of genes crossing-over by selecting

a subset of weights from each parent. A random mutation rate is also added to the weights at this point. The next generation comprises individuals with altered weights and the unaltered individuals that did not form part of the mating process. This process is repeated with the aim of reducing the error rate over a number of generations' (Whitley 1994)Pseudo code for the algorithm is shown in fig 4.4.

1. Apply Random weights to variables in training set
2. Select fittest individuals
3. Mate these individuals via crossover of their weights
4. Introduce mutation via random multiplier
5. Create new generation of altered and unaltered individuals
6. If exit condition not met, return to step 2

Figure 4-4 Genetic Algorithm Pseudo Code.

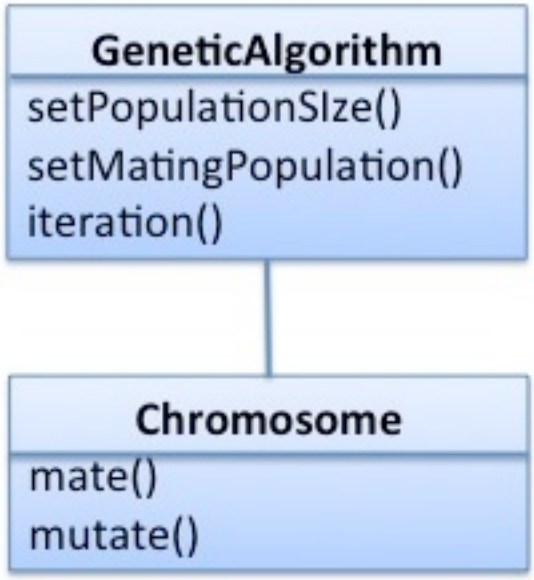

Figure 4-5 Partial UML (Unified Modelling Language) diagram of distinct classes required for the genetic algorithm technique together with their main methods

e) Utility classes

Various other classes have been utilised from the framework; the following table gives a brief summary of their uses:

Utility Class	Function
BiPolarUtil	Utilities for translating binary to bipolar (-1,1)
BoundNumbers	Simple checks to see if numbers within allowed limits.
ErrorCalculation	Calculates Root Mean Square errors
Matrix	A class to hold matrices of values
MatrixMath	Utility functions on matrices
NeuralNetworkError	Reports runtime errors from models
ActivationFunction	Interface for the different activation functions. Classes implementing these include linear, sigmoid and hyperbolic TANH
Delta	Calculates weights based on the delta rule

Table 4-1 List of Utility Classes required for modelling

4.2 Considered Modelling Techniques

The Encog framework provided a structure for performing a number of different types of machine learning models. The framework included six modelling techniques based on propagating values through the levels of the models and all of these techniques were tested. These techniques are described in section 4.2.1 through to section 4.2.6. Genetic Algorithms were also tested as they have the

different approach of selecting subsets of the population and this is described in section 4.2.7.

To observe the validity of the machine learning methods against an alternative approach to modelling, the results of the best performing models were also tested against a multiple linear regression model to determine their relative performance.

4.2.1 Backpropagation

The concept of momentum is especially important for backpropagation models as this type of modelling can suffer from local minima, where a small error is observed for a small subset of the data but does not maximise the global performance of the model. Momentum is provided as a parameter with a value of less than one and this slows the rate of amending weights. When a weight is to be adjusted in the same direction as on the previous iteration, the momentum value is multiplied by the learning rate to slow down the progress and make the model more unlikely to head down into local minima. For example, a momentum value of 0.5 would act to half the previous learning rate if the same errors were continually seen, this would ensure that a small group of similar observations would not be the driver of the entire model.

The technique of backpropagation was the original method of adjusting the weights of a feedforward network (Riedmiller 1994). On the forward stage,

calculated values are compared to actual values and the error gradient calculated. This gradient can be used to determine by what factor the learning rate should be applied to the weights via the gradient descent algorithm (Basheer & Hajmeer 2000). The concept of momentum is especially important for backpropagation models as this type of modelling can suffer from local minima, where a small local error is observed for a small subset of the data but does not maximise the global performance of the model. Momentum is provided as a parameter with a value of less than one and this slows the rate of amending weights. When a weight is to be adjusted in the same direction as on the previous iteration, the momentum value is multiplied by the learning rate to slow down the progress and make the model more unlikely to head down into local minima. For example, a momentum value of 0.5 would act to half the previous learning (Whitley 1994; Heaton 2010).

4.2.2 Quick Propagation

Another variant of backpropagation is that of Quick Propagation. This uses Newton's method (Fahlman 1988) instead of gradient descent to calculate adjustments. No momentum is required to use Quick Propagation and the method is generally more tolerant of larger learning rates. The models run more rapidly because of this hence the title Quick.

4.2.3 Manhattan Update Rule

The Manhattan Update rule (Schiffmann et al. 1993) amends the standard backpropagation model by amending weights by use of a constant rather than calculating a value by means of gradient descent. Manhattan models normally use very low learning rates, typically in the order of 1×10^{-4} %, and purely decide if the weights are too low or too high, they then add or subtract the learning rate as required.

4.2.4 Resilient Propagation

Resilient Propagation (Riedmiller & Braun 1993) takes the Manhattan Update Rule on to a further stage in that no parameters are required for learning rates and momentum. These are calculated for each individual weight and therefore allow a hugely flexible approach to amending the magnitudes of gradient descent.

4.2.5 Scaled Conjugate Gradient (SCG)

Conjugate Gradient Methods (Møller 1993) are another algorithm that can be used to optimise the rate of change of learning weights. It also does not require parameters to be set in advance and calculates the best amendments to be made based on the results it observes.

4.2.6 Levenberg Marquardt (LMA)

A hybrid of backpropagation and Resilient Propagation is the Levenberg-Marquardt Algorithm (LMA) (Ranganathan 2004). The main addition to the other forms of algorithm is that a variable damping factor is applied giving a more flexible approach to that provided by giving a model a momentum rate.

4.2.7 Genetic Algorithm

As discussed above, Genetic Algorithms require independent variables to be thought of as genes on a single chromosome. To simulate biological evolution, three phases form the role of weight adjustment after they have been randomly calculated:

a) Only the top performing observations are selected to breed, for example we may take the top 10% in terms of the lowest error rates and breed them on a random basis.

b) Crossover is simulated, a percentage of neuron weights (genes) are switched from one observation to another.

c) Mutation is simulated, a small number of weights have a random factor applied to them.

Some of the above models, such as LMA were known to have restricted performance on very large datasets of variables and observations (such as that extracted from the SETS database for this project) (Ranganathan 2004).

However, for completeness, all of the above seven techniques were tested and the results presented in Chapter 5.

4.3 Model Execution

The flowchart in figure 4-5 shows the four stages of model building and verification. Having examined the classes and techniques available, this section provides a walk through of the modelling process.

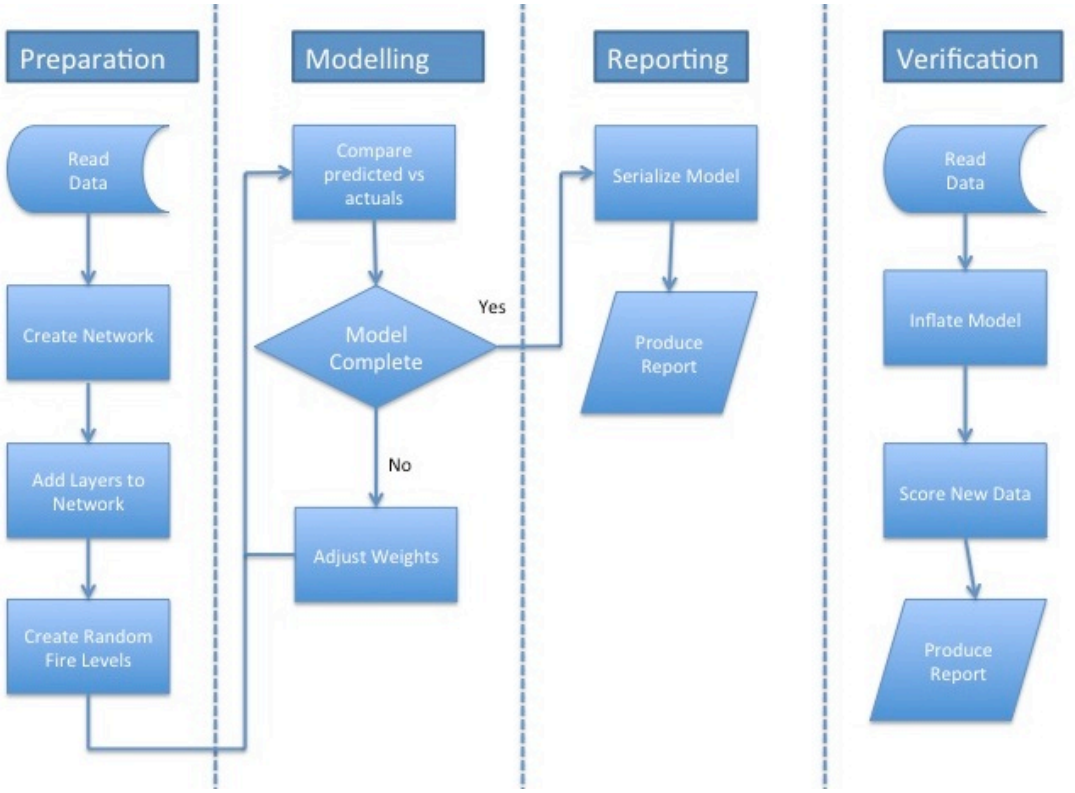

Figure 4-6 Flowchart summary of the required stages in producing machine learning models.

4.3.1 Preparation

Data are queried from the SETS database and used to populate the input layer of the model with the independent variables whilst the dependent variable is used to populate the output layer. A series of random values is generated to create initial matrices of weights that act as values for the hidden layer.

4.3.2 Modelling

Errors are computed between predicted and actual values to determine the accuracy of the activation levels. An acceptable level of performance in terms of predictive ability has been supplied as a parameter to the model; assuming that this hasn't been met and that the maximum iterations have not been reached, these weight levels can be adjusted. The adjustments are parameter driven, both the learning rate to control the percentage change at each iteration and, optionally, the momentum level to speed up or slow down the rate of change is provided at run time.

4.3.3 Reporting

Once the model has completed, the final values of the weights and activation values are extracted from the matrices, serialized and written to a file for use in the verification process. A report is also produced detailing the progress at each iteration and the final performance of the model.

4.3.4 Verification

In order to verify the model's accuracy it is important to test the results on data

that was not used in its creation. New data is firstly selected from the database based on random seeds. The serialised file is then reloaded and inflated back into matrices of scores that are applied to the new data. A report is then produced comparing predicted to observed accuracy.

The SETS database was stored in MySQL tables on the Sussex University High Performance Cluster (HPC) for speed and ease of access. Locally, both iMacs and a custom Linux machine were used for processing batches of models simultaneously.

This chapter has covered the design and production of the modelling environment. It has described the initial design through to code production reaching the point where the models could be executed. The results of the execution of the different models form the next chapter.

5 TFBS Modelling Results

Chapter four discussed the population of the SETS database and the production of all required classes for the modelling environment. With this data in place the testing of the predictive models was undertaken. The testing fell into two parts (a) the selection of the most effective modelling technique and (b) the adjustment of parameters to produce the best overall model for the most effective modelling technique.

This chapter examines the selection of the data, model and verification samples. It then presents initial results of the seven modelling techniques and the final results of the best modelling technique with optimally adjusted parameters.

5.1 Model Preparation

5.1.1 Sampling

As described in section 3.1 the SETS database holds details of 11,590,713 TFBS predictions from 98,751 transcripts based on position weight matrices. After these were merged with experimentally proven TFBSs and the independent data (see section 3.1), under-sampling was then used to create a table of data consisting of 142,438 observations. Of these 142,438 TFBS observations, 12,239 were experimentally verified and classified as true positives and 130,199 were classified as false positives. A sample of data was created to approximately match the number of true and false positives but in early testing did not provide as accurate models.

5.1.2 Selection of Model and Verification Samples

To prevent over fitting of modelling data, all output produced by the machine learning environment needed to be verified on data that was not used in the modelling process. From the available 142,438 observations, two distinct, randomly selected samples of records were taken, one for modelling and the other for verification purposes. During the first modelling stage, the size of these selections ranged from 2,000 to 25,000 records for the propagation models, and 500 to 10,000 for Genetic Algorithms. Smaller samples were used for Genetic Algorithms due to processing limitations. This process was performed independently for every model that was tested.

5.2 Parameter Adjustment

In machine learning models, various factors can affect the ability to produce a useful model. In addition to the number of input observations, additional parameters can be adjusted to optimise the observed results. These parameters vary depending on the modelling technique being used. Initial values were usually selected based on Encog (Heaton 2010) defaults with a process of trial and error determining the adjustments.

5.2.1 Backpropagation Model Parameters

For the six types of backpropagation models the adjustable parameters are shown in table 5-1.

Model Technique	**Learning Rate**	**Momentum**	**No of Hidden Neurons**	**Maximum No of Iterations of the model**
Backpropagation	Yes	Yes	Yes	Yes
Quick Propagation	Yes	No	Yes	Yes
Manhattan Update Rule	Yes – Very low	No	Yes	Yes
Resilient Propagation (RP)	No	No	Yes	Yes
Scaled Conjugate Gradient (SCG)	No	No	Yes	Yes
Levenberg Marquardt Algorithm (LMA)	No	No – automatically calculates damping rates	Yes	Yes

Table 5-1 Adjustable parameters by modelling technique.

The parameters shown in table 5-1 consist of:

a) Learning rate: The learning rate is a percentage figure and is the rate at which weights are adjusted up or down after each iteration. These rates are typically small fractions of a percentage such as 0.0001, as seen in table 5-2. These low learning rates help to prevent uncontrollable swings in the modelling process that would inhibit the model converging on a low error rate.

b) Momentum: Momentum is a factor that is applied to the learning rate of models in order to prevent the model converging on local minima rather than producing a more accurate model. It is typically a large percentage (e.g. 50%) of the learning rate, and reduces the amount of rate of change that is applied.

c) Number of hidden neurons: The number of neurons or synapses that are created in the hidden layer is an adjustable parameter that can have considerable affect on results. Too many neurons and the model can describe the data but not produce good results on the verification set. Too few neurons and the model will not have enough flexibility to produce a good solution.

d) Maximum number of iterations of the model: A model will cease running when either the desired error rate is reached, or if it performs a certain maximum number of iterations.

5.2.2 Genetic Algorithm Parameters

Genetic Algorithms form a different class of machine learning models and therefore require a different set of parameters to be adjusted and set. As has been seen in Chapter 1, the variables can be thought of as genes on a single chromosome and their weights can be considered as regulatory features that determine if the gene is transcribed or not. Therefore the following parameters are the ones that have been adjusted in the test matrices (table 5-2):

a) Mating Population: The observations that most accurately predict the output are those that are selected to breed and form the next generation. This is an

adjustable proportion of the dataset, e.g. the top 25% of the population may be chosen to breed.

b) Mutation rate: A degree of random mutation is applied to the weights of the variables of the selected breeding observations.

c) Number of generations: As an exact representation of the maximum number of iterations of the model, the maximum number of generations can be specified.

5.3 Measurement Criteria

Before the outset of any modelling exercise it is important to consider how the performance of each modelling technique will be assessed. The performance of the training set is measured by the error rate of the model. The actual validity of the model is measured using the ratio of accurately predicted true positives in the verification set.

5.3.1 Error Rate

As the system performs the modelling on the training set it measures the error rate, the percentage of misclassified observations, at the end of each iteration. This error rate is generally reduced over the course of many iterations until the model accurately describes the data to a predetermined level, typically 97.5% accuracy or an error rate of 2.5%. This error rate will not represent the actual accuracy of the model. In datasets with many variables, such as those in the

current work, it will generally considerably over-estimate its precision (Smith 2006). It is therefore important to measure the actual performance by scoring the verification set and producing reports such as ROC curves to assess model validity.

5.3.2 ROC Curves

Receiver operating characteristics (ROC) curves are so called because they graphically compare the two operating characteristics of true positive rate (TPR) against false positive rate (FPR) (Robertson & Zweig 1981). As they represent a 2 x 2 contingency table as shown in figure 5.1, they allow the effectiveness of the model to be shown graphically. The two key measures illustrated in a ROC curve are (where TP = true positives, FP = false positives, TN = true negatives, and FN = false negatives):

(a) Sensitivity: SEN = TP / (TP + FN)

b) Specificity: SPE = TN / (TN + FP)

A ROC curve is 1.00 – SPE plotted on X-axis against SEN plotted on the Y-axis. The most common summary statistic presented alongside ROC curves is the area under the curve (AUC) statistic (Hanley 1982). This has been used to compare the performance of models on the verification sets. The ROC curve and AUC measurements were calculated by the ROCR R package (Sing et al. 2005).

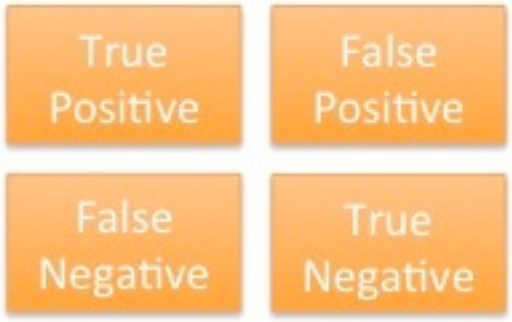

Figure 5-1 2 x 2 contingency table showing the four states represented in a ROC curve plot. The area under the curve (AUC) measures how predictive the model is.

5.4 Initial Results

The initial stage involved the investigation into the effectiveness of different techniques. Results were initially compared in terms of how well the models performed via the reduction in error rate. If the target of reducing the error rate, the percentage of misclassified outputs, to 2.5% was met, scoring the verification set was then used to check the validity of the model.

5.4.1 Determination of Test Matrices

The adjustable parameters detailed in section 5.2 were combined in conjunction with the size (in terms of observations) of the training sets to produce a test matrix. In total 187 models were produced for this first stage to test the different modelling techniques with a range of parameters as shown in tables 5-2 and 5-3.

Model Technique	No of Obs modelled	Learning Rate	Momentum	No of Hidden Neurons	Maximum No of Iterations of the model
Backpropagation	2,000 - 25,000	5 x10^{-5} – 0.01	0.01 – 0.75	18 - 42	10,000 – 1,000,000
Quick Propagation	2,000 – 25,000	0.01 – 1.00	na	18 - 42	10,000 – 1,000,000
Manhattan Update Rule	2,000 - 25000	1 x 10^{-6} – 5 x 10^{-6}	na	18 – 30	10,000 – 500,000
Resilient Propagation	2,000	na	na	18 – 30	5,000 – 10,000
SCG	2,000 – 25,000	na	Na	18 – 30	10,000 – 100,000
LMA	200 – 2000	na	na	18 – 30	5000 – 1,000

Table 5-2 Range of parameters tested by backpropagation models.

Model Technique	Population Size	% of pop to breed	Mutation Rate
Genetic Algorithm	500-10,000	0.1 – 0.25	0.01 – 0.2

Table 5-3 Range of parameters tested by Genetic Algorithm models.

The two largest factors in execution time of the model were (a) the number of observations used in the training set and (b) the maximum number of iterations allowed before the model terminated. The number of hidden neurons also contributed to the run times but, along with learning rate and momentum, were able to be tested over a large range.

5.4.2 Modelling Technique Comparisons

The comparison of modelling techniques was performed on three groups based on the size of the training set. For propagation techniques the groups were 2,000, 15,000 and 25,000 observations whilst Genetic Algorithms were tested on population sizes of 500, 2000 and 10,000. Genetic Algorithms were tested on a training set of 500, 2000 and 10,000. The smaller Genetic Algorithm training set sizes were required due to both time and memory limitations when very large datasets were used.

a) Small group size (2,000 and 500 observations)

The initial selection size was limited to 2,000 observations for the propagation techniques and a population of 500 for the Genetic Algorithms. Although these models with limited dataset size did not perform well on the verification set, they provided early indications of the potential of the different modelling techniques. The results achieved in terms of error rates are shown graphically in figures 5-2 and figure 5-3.

Two techniques, those of LMA and Resilient Propagation, would not converge even with the smallest values of parameters applied. Therefore it was concluded at this early stage that the best results would not be obtainable using these methods. Backpropagation produced good early results with a mean error rate of 6.47% over 34 test models whilst Quick Propagation produced the best mean performance with an error rate of 6.2% over 11 different test models. The Scaled Conjugate Gradient models performed very consistently with a mean

error rate of 7.75% and the smallest IQR (Inter Quartile Range) 0.475%. Manhattan models produced the worst results of the propagation techniques on the smaller dataset achieving a mean error rate of 12.37%. Genetic Algorithm initial tests were based on a population size of 500 and produced a mean error rate of 11.5% within a small IQR of 1.675%.

The Resilient Propagation technique, having considerable flexibility in terms of calculating individual learning rates for each input variable (Riedmiller & Braun 1993), was more appropriate for smaller datasets and hence did not converge. The LMA technique being a hybrid of Resilient Propagation and Backpropagation (Ranganathan 2004) suffered from the same issue and also didn't converge. Due to the lack of convergence on the smaller models, LMA and Resilient Propagation techniques were ruled out as unsuitable at this stage. The other five modelling techniques were then tested on larger datasets.

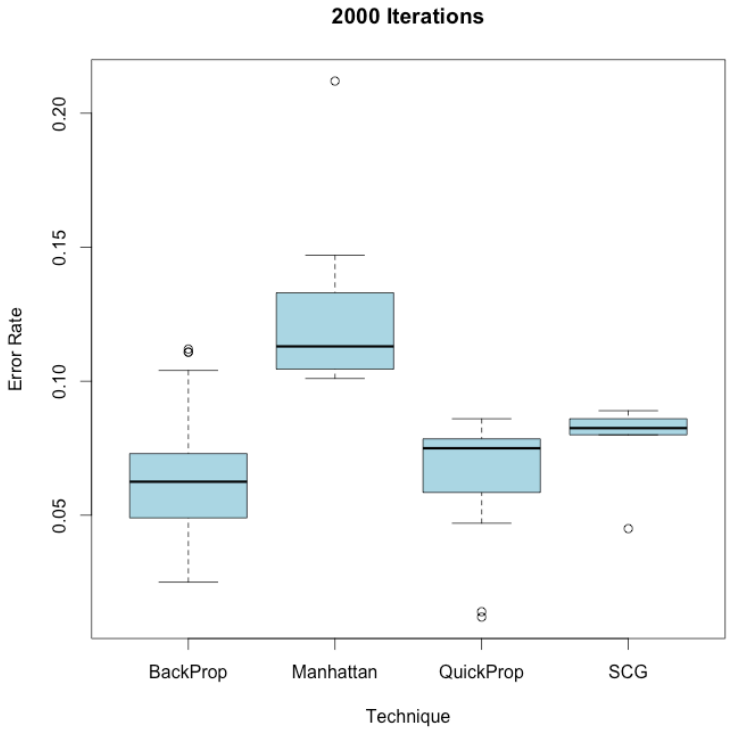

Figure 5-2 Box-whisker plot of error rates observed using 2,000 observations.

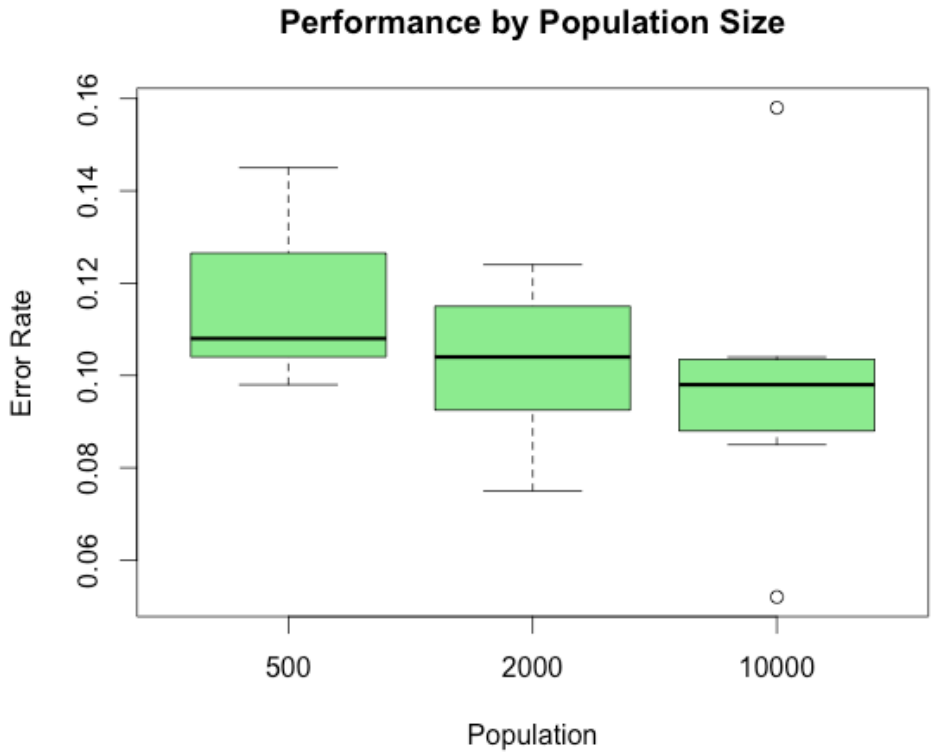

Figure 5-3 Genetic Algorithm error rates by population size.

b) Mid-sized groups (15,000 and 2,000 observations)

The expansion to mid-sized groups with 15,000 observations produced reduced error rates in three of the four tested propagation methods as shown in Figure 5.4. The exception was the results of the Scaled Conjugate Gradient models where the mean error rate rose slightly to 8.03%. Manhattan models improved to a mean error rate of 11.8%. The most successful models belonged to backpropagation, with a mean error rate of 3.17% and an IQR of 0.153%, and Quick Propagation with a mean error of 2.93% but a larger IQR of 0.447%.

The Genetic Algorithm results seen in figure 5-3 show mean error rates reducing to 10.28% for the increased population size of 2000.

All of these five techniques were then subjected to tests with the larger datasets.

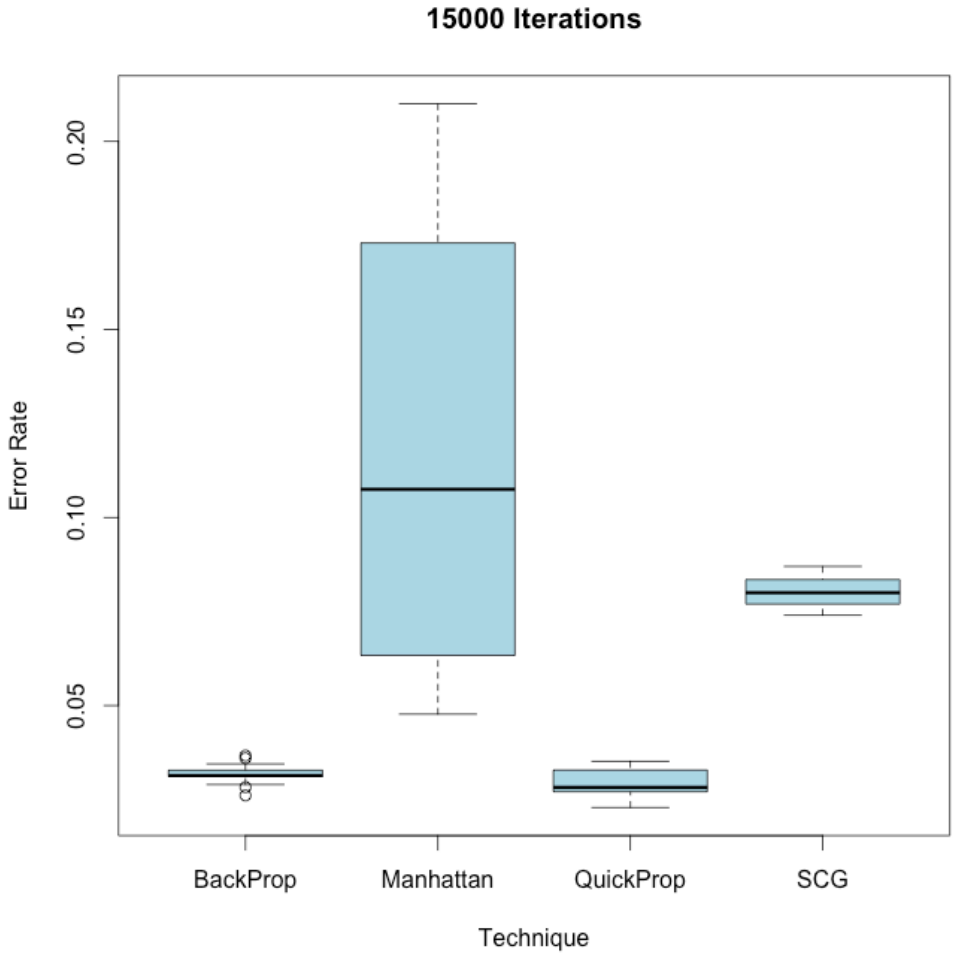

Figure 5-4 Box-whisker plot of error rates observed using 15,000 observations.

c) Largest groups (25,000 and 10,000 observations)

The largest datasets tested consisted of 25,000 observations for the propagation methods (Figure 5.5). The increase in size again had little effect of the Scaled Conjugate Gradient models that achieved a mean error rate of 7.1% with an IQR of 1.65%. Manhattan models saw a limited improvement resulting in a mean error rate of 10.08%.

The maximum size of population that was achievable for Genetic Algorithm models was a population size of 10,000 due to both long run times and high memory requirements. As shown in Figure 5-4, the mean error rate reduced to 9.87% with a consistent IQR of 1.55%.

The best results were again seen with the backpropagation models, a mean error rate of 3.35% and an IQR of 1.07%, and Quick Propagation where a mean of 3.76% and an IQR of 1.67% were observed.

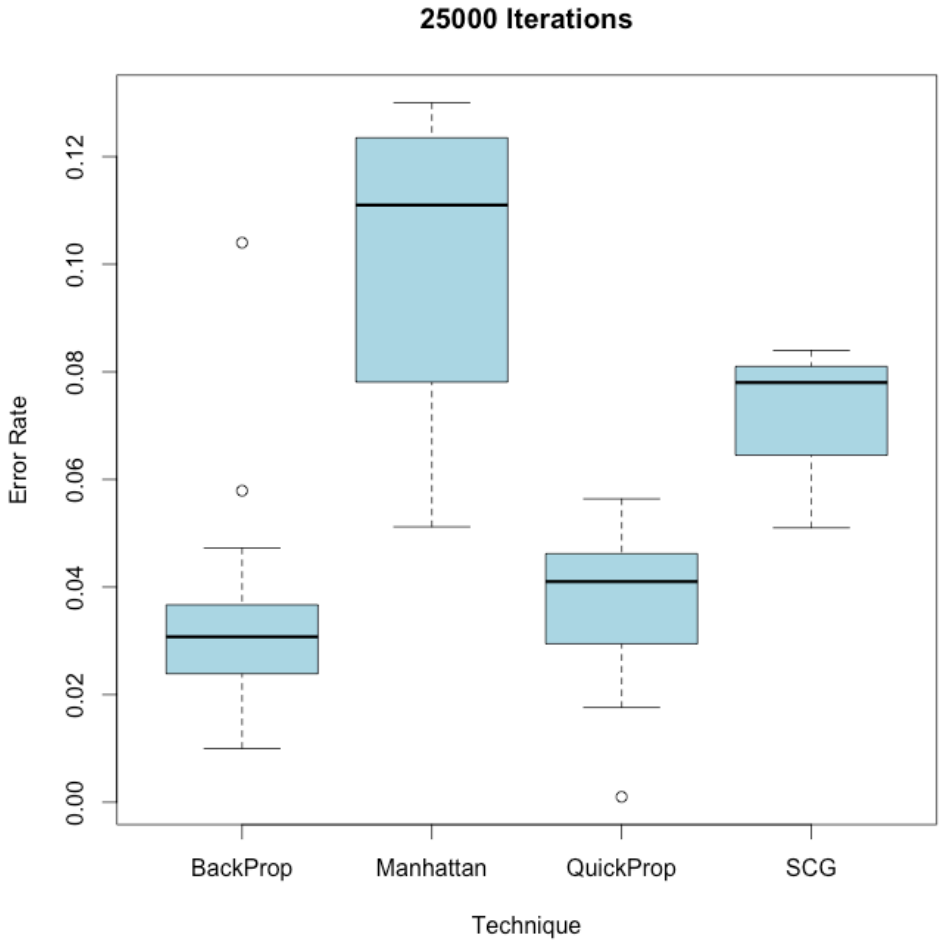

Figure 5-5 Box-whisker plot of error rates observed using 25,000 observations.

5.4.3 Selection of Best Technique

Looking at the results above it was apparent that only the backpropagation and Quick Propagation techniques were capable of producing the highest performing models. The best performing Manhattan, SCG and Genetic Algorithm models were tested but achieved verified results of 54.1%, 62.1%, and 50.7% respectively.

The backpropagation and Quick Propagation models that achieved the target of a 2.5% or less error were all verified against an independent sample. Backpropagation achieved a mean verification AUC of 69.93% with an IQR of 4.26% and Quick Propagation models saw a mean of 65.2% with an IQR of 5.35%. Although Backpropagation and Quick Propagation produced similar results, Backpropagation produced more consistent results with a higher mean AUC and hence it was selected as the best technique and taken forward for further optimisation in order to maximise the accuracy of the final model.

5.5 Final Results from Backpropagation Modelling

Having selected backpropagation as the preferred technique, 123 further models were created, each using a different independent sample of 25,000 observations. Parameters were amended based on the results of the initial modelling.

5.5.1 Parameters used in Final Modelling

The best performing models of the initial modelling resulted in the following parameter limits being tested for the final models:

a) Learning Rate: Range of values between 1×10^{-6} and 1.6×10^{-4}

b) Momentum: Range of values between 7×10^{-6} and 1×10^{-4}

c) Number of Hidden neurons: Range of values between 30 and 40

d) Number of Iterations: Range of values between 100,000 and 2,000,000

5.5.2 Model Comparisons

A total of 123 different models were created for this phase. The increased variation in the parameters led to an increased average error rate of 3.96%, however the best performing model resulted in an error rate of 1.80%.

The performance by variation in parameters is shown in figure 5-6. The best performing learning rates tended to be in the mid range with best results using 4 x 10^{-6} that produced a mean error rate of 3.61%. Momentum performed better with smaller values with the lowest mean error rate of 3.43% seen at 6 x 10^{-5}. The variation in number of hidden neurons did not have a major effect on results but the value of 38 performed best at an average of 3.63%. The major increase however was seen in the number of iterations, by increasing the upper limit to 2,000,000 the mean error rate averaged 3.40%.

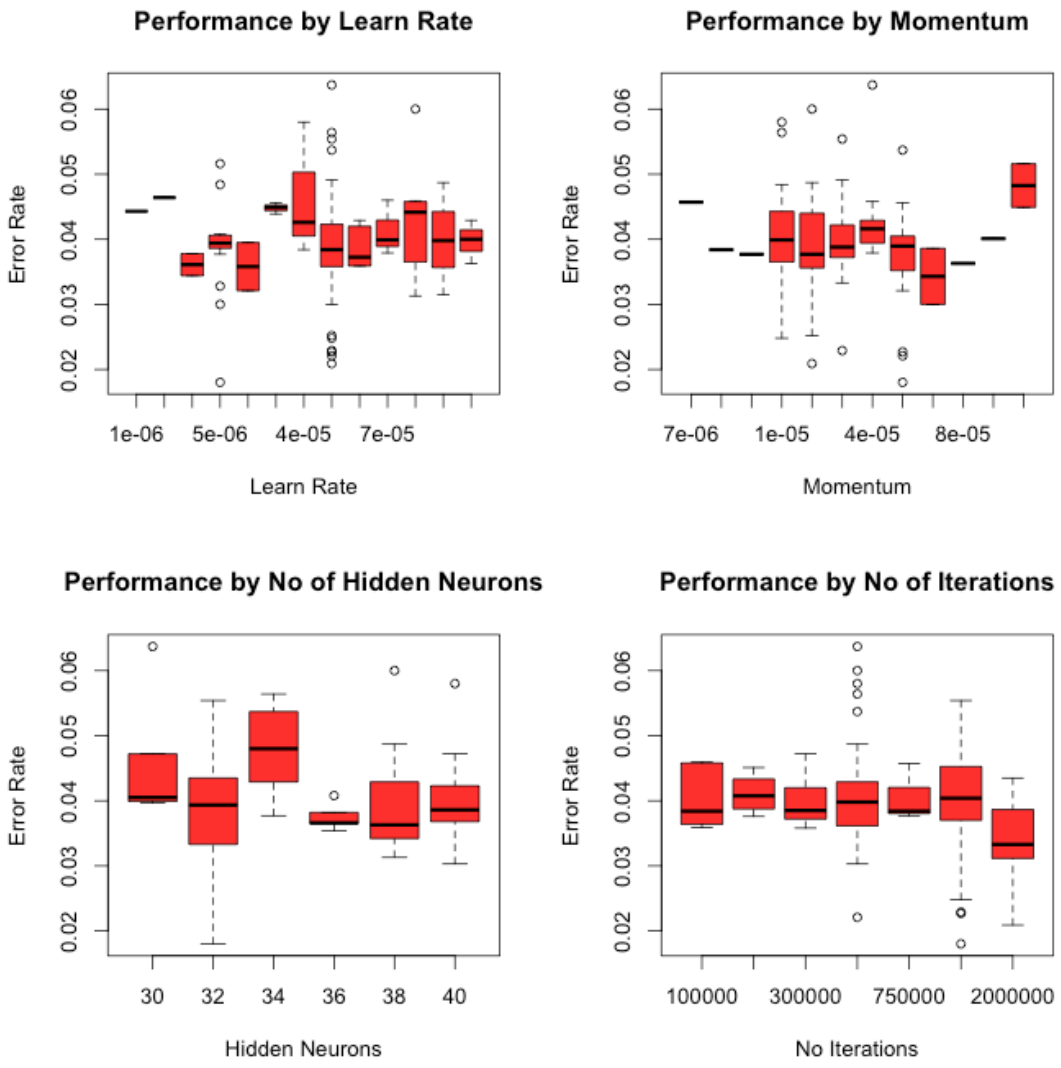

Figure 5-6 Backpropagation model performance by parameter variation.

5.5.3 Verification of Final Model

The individual model that produced the best result in terms of error rate consisted of the following parameters:

Learning Rate:	5 x 10-6
Momentum:	5 x 10-5
Hidden Neurons:	32
Iterations:	1,000,000

All of the models that produced an error rate of 2.5% or less were verified and resulted in mean AUCs between 75% and 82%. The top-performing model was verified on three different samples and the results merged. This resulted in an observed mean AUC of 79.9% as seen in Figure 5-7.

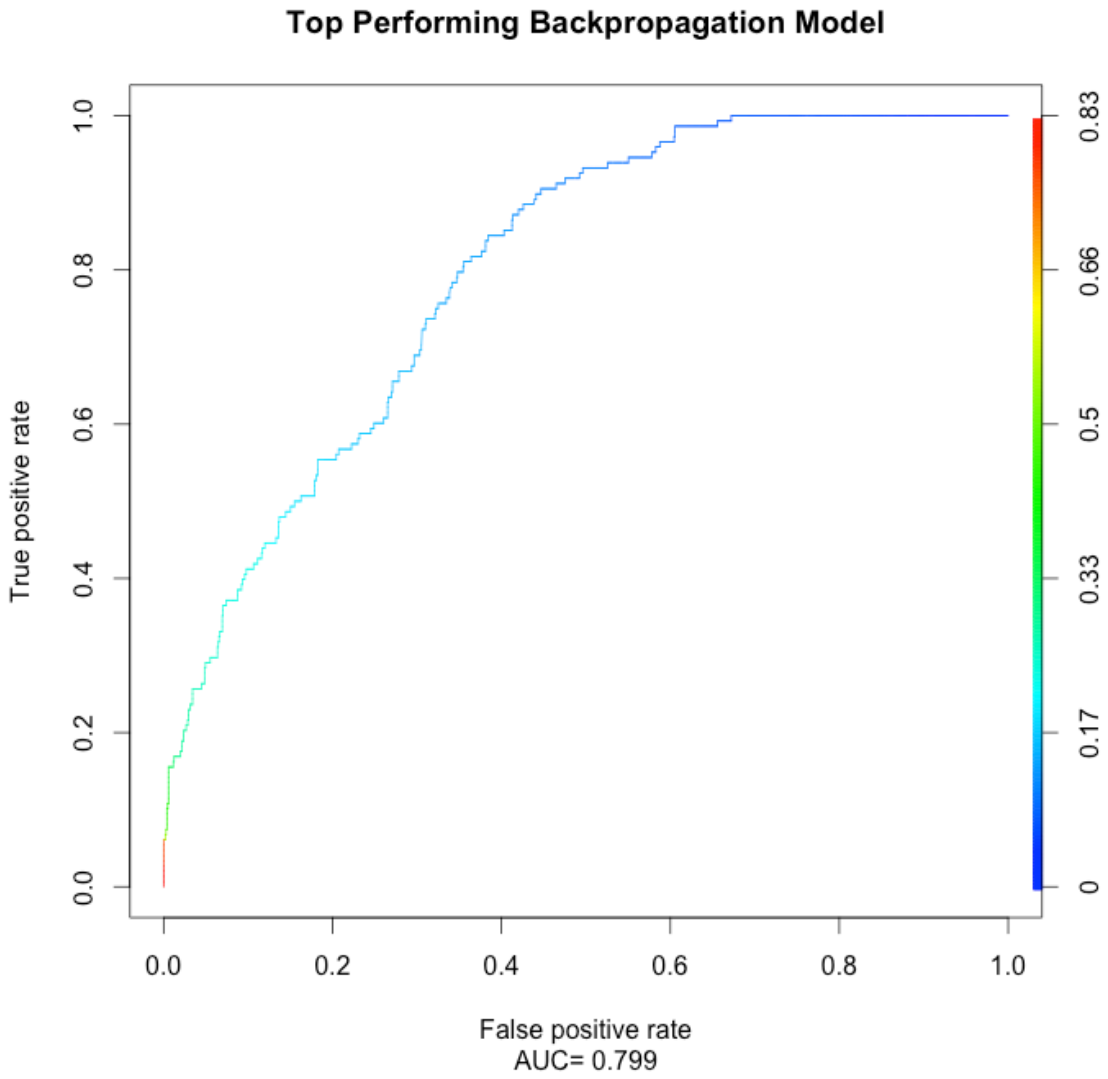

Figure 5-7 ROC curve of top performing backpropagation model. AUC is represented by the right hand Y-axis and by hot to cold colours.

5.6 Multiple Linear Regression Comparison

In order to determine and compare the efficacy of machine learning models overall, it was decided to compare the best results with those achievable from the more traditional technique of linear regression.

Two randomised 25,000 observation datasets were randomly selected from the SETS database to create modelling and verification samples. All of the same data were used as independent variables with the TFBS being experimentally verified as a binary dependent variable. The model was produced in the R language by the lm (Linear Model) function and the process was repeated three times with different random selections of data. The observed AUC results on the verified files were 0.731, 0.735 and 0.756.

The linear regression model did outperform many of the earlier machine learning models (see section 5.4.3). However, the highest observed AUC of 0.756 (figure 5.8), was smaller than that achieved by the best performing backpropagation model (AUC 0.799). Linear Regression was also tried on the dataset excluding the entropy variable. Although the entropy variable had a coefficient of 0.00986 on the previously best performing model, excluding it did not change the AUC from 0.756.

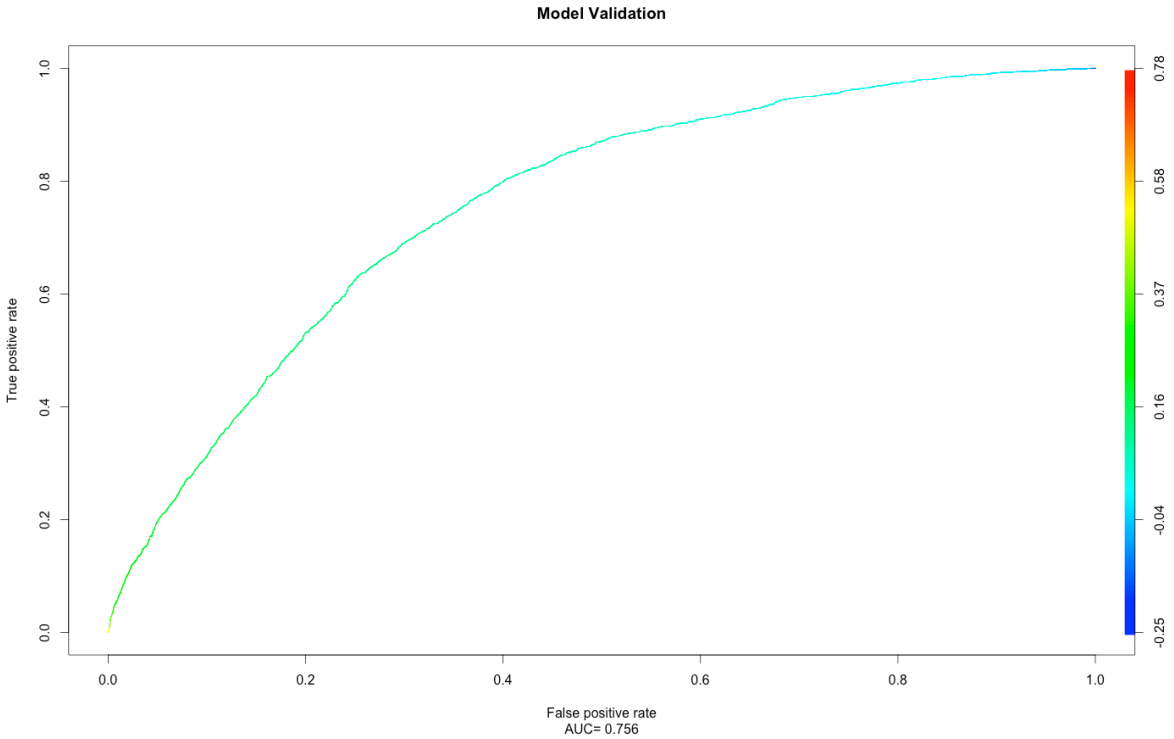

Figure 5-8 ROC curve of best performing multiple linear regression model applied to the verification dataset. AUC is represented by the right hand Y-axis and by hot to cold colours.

5.7 Modelling TFBS for the complete dataset

The final model, having been validated against 25,000 observation samples, then needed to be applied to the complete dataset in order to determine the level of similarity between experimentally proven TFBSs and those predicted by PWMs. The complete dataset comprised the 11,590,713 TFBS predictions held in the SETS database however these needed to be restricted to those that matched all variables required for the scoring. The main restricting factor was that expression levels in the various cell types needed to be available, which resulted in a universe of 5,035,802 records that could be scored. The breakdown of scores is shown in Table 5-4.

Predicted Score	Observation Count	Cumulative Count
0.901-1.000	9569	9569
0.801-0.900	7569	17138
0.701-0.800	8343	25481
0.601-0.700	10394	35875
0.501-0.600	13567	49442
0.401-0.500	15944	65386
0.301-0.400	24803	90189
0.201-0.300	44249	134438
0.101-0.200	98749	233187
0.000-1.000	4802615	5035802

Table 5-4 - Breakdown of model scores applied to the universe of predictable TFBSs.

As with PWMs, the score represents relative fit with the model, for example, a score of 0.8 represents 80% similarity with the characteristics of an experimentally verified TFBS. As shown in table 5-4, scores of 0.8 or higher produce 17,138 TFBSs with 80% or higher similarity to those experimentally proven. These most similar TFBSs have been combined with the experimentally verified ones from ENCODE and used to conduct the CRM predictions in Chapter 6.

5.8 Discussion

This chapter detailed the testing and selection of the most efficient models. Although even the best results show a considerable number of TFBSs are being misclassified, the achieved AUC of 79.9% is a dramatic improvement over published futility theorem figure (see Chapter 1) of 1000 false positives for each true positive.

In comparable work, five TFBS location techniques have been compared using a limited benchmark dataset comprising TP and FP TFBS data for nine transcription factors (Handstad 2011). This work used the ENCODE (Birney et al. 2007) ChIP-Seq data on nine TFs as their true positives and random DNA as their true negatives. The five techniques of standard PWM search, MotifScan (Naughton et al. 2006), weighted sum (a method based on sequence conservation), Bayesian branch length score (BBLS) (Xie et al. 2009)and a combined approach (BBLS+MS) (developed by the paper's authors) were tested against the benchmark dataset. Although the model based on the SETS database cannot be directly compared and applied to the Handstad dataset as the SETS database used different transcription factors, was based on 16 rather than 9 transcription factors and furthermore looked at different upstream regions, it is possible to compare the results in terms of AUC of the ROC curves. The results they achieved, in terms of median ROC AUC, varied from 70.01% for their combined method, through to 72.51% for those identified by MotifScan were exceeded by the mean 79.9% AUC achieved by the SETS database model.

The next chapter presents a method for applying the best backpropagation techniques developed here, to the identification of TFs that potentially act in *Cis*-

regulatory modules in genes within linkage disequilibrium blocks that potentially are related to a specific disease state or trait.

6 Investigation into potential *cis*-regulatory modules using data from Genome Wide Association Studies

In order to examine the effectiveness of the models, both the experimentally verified TFBSs and those predicted with the highest model scores have been applied to publically available data. The largest and most relevant available data are those results from Genome Wide Association Studies (GWAS). These studies examine differences in DNA between two populations, those exhibiting a trait compared to those without. GWASs became possible due to the combined availability of the human genome sequence, haplotypes from various world populations provided by the International HapMap Project (HAWKS 2005), and millions of publically available single nucleotide polymorphisms (SNPs). GWASs are performed by comparing SNP alleles between groups exhibiting a phenotypic difference. An association is found if there is a statistical significant correlation between the genotype and the phenotype (Hirschhorn & Daly 2005).

The GWAS catalog (Hindorff et al. 2011) (accessed on 7/10/2013) consists of 1,750 published experiments assaying at least 100,000 SNPs. Results are provided for SNPs associated with a variety of traits with p-values $< 1.0 \times 10^{-5}$. The database is maintained by the US National Institute of Health (NIH) and has been downloaded in full.

The GWAS catalog contains data of the id (rs number) of the SNP but does not contain genomic coordinates; furthermore it does not distinguish the chromosomal regions or linkage disequilibrium blocks that are associated with a

given trait. To rectify this, data from the DistiLD database (Pallejà et al. 2012) has been downloaded and merged. This database uses data from the International HapMap program (HAWKS 2005) to partition each chromosome into linkage disequilibrium blocks. The International HapMap program analysed over 1 million SNPs from various world populations looking at which SNPs were inherited together. These co-inherited areas of the genome containing multiple SNPs were classified into linkage disequilibrium blocks. The genomic coordinates of each block can be analysed to associate various genes to all of the SNPs that appear within that block, thereby permitting the analysis of phenotypic traits by any of the associated genes.

6.1 Production of Dataset

To extract a dataset to apply the TFBS information to, data was extracted and merged from the DistiLD database, the genome catalog GWAS database, and true and top ranking false positives from the SETS database.

6.1.1 TFBSs

The complete SETS database contains 5,035,802 scored TFBSs, the top 17,138 (0.34%) of those having the highest predicted propensity (a score of >= 0.80) to be true positives (see Chapter 5.7). These were combined with the 29,095 experimentally verified TFBSs (from ENCODE (Dunham et al. 2012)) to create 46,233 observations. These TFBS observations were then aggregated at the gene level to produce a list of all TFBSs observed in the promoter of a particular gene.

TFBSs observed to be conserved in specific combinations at specific locations within the promoters represent potential CRMs, regulating the expression of genes in a linkage disequilibrium block related to a specific trait or disease phenotype. Due to the very high numbers of potential permutations, the TFBSs were treated as binary outcomes for initial reporting purposes. For example, although multiple AP1 TFBSs (the activator protein 1 associated with cellular processes including apoptosis (Wasserman & Sandelin 2004))may be seen in a gene promoter, this counted as having an AP1 TFBS. Additionally, although the order of these TFBSs is important, this initial analysis examined the presence or absence of TFBSs rather than a specific order. This resulted in 403 observed combinations ranging from single TFBSs to a combination of 14 different TFBSs as shown in table 6-1.

No of TFBSs in Promoter regions	No of Traits Associations in LD Blocks
1	1200
2	1547
3	1262
4	797
5	373
6	227
7	123
8	121
9	98
10	142
11	116
12	68
13	37
14	10

Table 6-1 No of different TFBSs observed in 1500 bp upstream to 200bp downstream in gene counts against traits associated within LD blocks.

6.1.2 ICD-10 codes

Phenotypes used by the GWAS catalog are coded by the 10th revision of the International Statistical Classification of Diseases and Related Health Problems (ICD-10) (International Classification of Diseases) provided by the World Health Organization (WHO). This is a hierarchic code with the first three digits representing the class, for example, the code A37 represents Whooping cough with A37.0, A37.1, A37.8 and A37.9 representing specific strains.

At the three digit level, 205 different ICD-10 codes were observed in the GWAS catalog and merged to create the analysis dataset. A default code of "trait" was observed in the dataset relating to unclassified phenotypes and was ignored for reporting purposes. The traits at the three-digit level with >= 100 observations are shown in table 6-2.

ICD-10	Description	No of Studies
F31	Bipolar disorder	361
E11	Type II Diabetes Mellitus	337
M05	Rheumatoid arthritis	258
G30	Alzheimer's disease	207
G20	Parkinson's disease	188
G12	Sporadic Amyotrophic Lateral	159
I25	Coronary heart disease	151
F20	Schizophrenia	131
E66	Obesity-related traits	124
J44	Spirometric measures of lung	121
K90	Celiac disease	119
G35	Multiple sclerosis	116
K50	Crohns disease	111
K51	Ulcerative colitis	105
I10	Blood pressure	100

Table 6-2 ICD-10 codes and descriptions with number of GWAS. Table limited to ICD-10 codes with 100 studies or more.

6.1.3 Genes and Linkage Disequilibrium Blocks

The DistiLD database (Pallejà et al. 2012) (accessed on 7/10/13) consists of 37,989 linkage blocks although only 5,274 of those contain genes. At the gene level, 3,196 genes were associated with SNPs from the GWAS catalog and these were found in 1,663 different linkage disequilibrium blocks.

These three sets of data were merged to produce 21,650 records each containing, a gene, a LD block, a phenotype and observed TFBSs (predicted and

experimental).

6.2 Methods to examine the significance of TFBS Combinations

Reports were created from this dataset looking at overrepresentation and significance of the TFBSs within particular ICD-10 traits.

6.2.1 Calculation of Frequencies

The initial stage of reporting was the creation of a frequency table of counts by TFBS combinations and ICD-10 code created by SQL. This table then had an observed and expected value calculated based on the overall frequencies of the TFBS combinations and the ICD-10 code. A Pearson chi-squared value was calculated using the standard formula:

$$X^2 = \sum \frac{(O - E)^2}{E}$$

Where O = observed and E = expected.

6.2.2 Production of Contingency Tables

In order to check the statistical validity of the results by adding p-value to the

single chi-squared statistic, code was written in the R language (Team 2005)to create 2 x 2 contingency tables for each combination of TFBSs and ICD-10 code. Of the potential 82,615 combinations (403 TFBS combinations by 205 ICD-10 code) combinations, 6,895 had at least one observation and contingency tables for these were produced for analysis in R.

The R chisq.test was used to calculate both the contingency table chi-squared value and the p-value. As an assumption of the chi-squared statistic is that it is not reliable where expected values are under 5 (Plackett 1983), the R fisher.test was used to perform the Fisher Exact test on these smaller samples. Both the chi-squared and Fisher Exact tests were calculated with 1 degree of freedom due to the use of the 2 x 2 contingency tables.

6.2.3 Bonferroni Correction

As many hypotheses were being simultaneously tested using the contingency table approach, corrections needed to be applied to handle the issue of multiple testing. Bonferroni corrections (Cabin & Mitchell 2000) are the most common approach to this problem as they are simple to apply and also thought of as conservative.

The Bonferroni correction applied was:

$$s = \frac{\alpha}{n}$$

Where s is the corrected significance level, α is the base significance level, and n is the number of concurrent tests.

6.2.4 Bootstrapping

To further investigate the validity of results, the bootstrapping technique has been employed (Grimshaw et al. 1995). Bootstrapping is a method of resampling where multiple random, independent copies of data are resampled in order to produce a distribution. This distribution can then be used for hypothesis testing by means of calculating confidence intervals for the random samples and comparing them to observed results.

For the observed counts of number of genes for each ICD-10 code, a random selection of genes was taken from the complete dataset and the number of different TFBS combinations observed recorded. Once this process was repeated 1000 times the distribution of counts was then calculated and compared to the actual observed counts of TFBS combinations. This 1000 times sampling was then repeated to produce a further check on observed results. Finally, for the lowest 10 p-values in both the chi-squared and Fisher's Exact groups, 25,000 bootstrapped samples were performed.

P-values could then be created based on the probability of falling within the randomly produced distributions.

6.3 Results

Reports were created in two stages (a) a comparison of observed against expected values was made, and then (b) these results were validated via independent runs of 2,000 and 25,000 random selections obtained using the bootstrapping technique.

6.3.1 Observed vs. Expected Tables

To examine the statistical significance of data and produce a p-value, initial results needed to be split into two sections as shown in 6.2.2, a combination of both chi-squared and Fisher's Exact tests techniques were therefore performed.

As multiple comparisons were made, Bonferroni corrections were applied to both studies. There were 584 chi-squared comparisons and 1584 Fisher's exact comparisons. A base level of significance of 0.05 was applied resulting in the following adjusted p-values for significance:

- Chi-Square test – 0.05 / 584 = 8.56164 x 10-5
- Fisher's Exact test – 0.05 / 1584 = 3.15657 x 10-5

The Bonferroni adjusted p-values for significance are thought to be conservative (Cabin & Mitchell 2000). In the chi-squared tables (see figure 6-3), despite large differences between observed and expected, no values had a p-value of less than 8.56×10^{-5} and hence could be considered significant.

The Fisher's Exact test group did produce three significant values taking into account the Bonferroni corrections. Combinations of 4, 2, and single TFBSs were

observed multiple times in the conditions described by the ICD-10 codes as, Optic disc parameters (H40), Venous thromboembolism (I26), and Haemolytic anaemia in hepatitis (D59). Each set of TFBSs predicted to occur in genes linked to each condition had significant p-values of less then 3.15×10^{-5}.

TFBS combination	ICD10	Description	Obs	Expected	Chi-Squared	p-value
USF1	G35	Multiple sclerosis	20	9.38328	11.38	0.0007
NFYA,TFAP2A	K50	Crohns disease	12	5.05788	8.49	0.0036
NFYA	I25	Coronary heart disease	67	51.38633	4.97	0.0258
CTCF	J44	Spirometric measures of lung	75	93.22956	4.39	0.0361
NFYA	G12	Sporadic Amyotrophic Lateral S	73	58.05182	4.04	0.0445
CTCF	C43	Melanoma	6	12.79196	3.96	0.0465
CTCF,E2F1	G30	Alzheimer's disease	4	10.57339	3.72	0.0538
NFYA	M45	Ankylosing spondylitis	5	11.54836	3.45	0.0630
SPI1	M32	Systemic Lupus Erythematosus	13	7.54975	3.39	0.0652
CTCF	Q35	Nonsyndromic cleft lip	10	5.63714	3.38	0.0658
CTCF	R72	Hematological parameters	23	16.26097	3.07	0.0799
E2F1	M05	Rheumatoid arthritis	51	40.10383	2.98	0.0843
CTCF	C91	Acute lymphoblastic leukemia	29	21.46448	2.96	0.0855
CTCF	K80	Gallstones	35	45.53072	2.85	0.0914
CTCF	M81	Bone mineral density	40	31.22106	2.82	0.0930

Table 6-3 Table of Observed vs. Expected values of ICD-10 codes by TFBS combination. Top 15 values, Chi-squared test used, expected values 5+

TFBS Combination	ICD	Description	Obs	Expected	Fishers	p-value
E2F1,MAX,NFYA,SRF	H40	Optic disc parameters	3	0.03427	178.33	**3.94E-06**
CTCF,NFKB1	I26	Venous thromboembolism	2	0.00443	504.94	**7.67E-06**
NR1H2::RXRA	D59	Haemolytic anaemia in hepatitis	2	0.00739	302.07	**1.99E-05**
AP1,CTCF,MAX,MYC::MAX,SP1,SPI1,TFAP2A,USF1,YY1	I45	Electrocardiographic traits	2	0.00878	253.73	3.27E-05
NFYA	Q23	Aortic root size	6	0.77506	31.24	3.43E-05
MAX	C44	Cutaneous basal cell carcinoma	3	0.10088	58.05	9.08E-05
SP1,USF1	I48	Atrial fibrillation	2	0.01829	120.29	0.0001
E2F1,SP1,SPI1	B24	HIV-1 disease progression	3	0.11778	48.78	0.0002
CTCF,E2F1,MAX,SP1	C16	Diffuse-type gastric cancer	2	0.02263	96.70	0.0002
E2F1,MAX,NFYA,SPI1,USF1	C16	Diffuse-type gastric cancer	2	0.02263	96.70	0.0002
CTCF,NFYA,SRF,YY1	Y44	Antiplatelet Effect and Clinic	2	0.02457	88.93	0.0003
MAX,RXRA::VDR,TFAP2A,USF1	Y44	Antiplatelet Effect and Clinic	2	0.02457	88.93	0.0003
AP1,MAX	I70	Subclinical atherosclerosis	4	0.315935335	32.41	0.0003
AP1,MAX	I45	Electrocardiographic traits	2	0.03334873	64.67	0.0005
CTCF,NFYA,SP1,TFAP2A	Y44	Antiplatelet Effect and Clinic	2	0.035103926	61.37	0.0005

Table 6-4 Table of Observed vs. Expected values of ICD-10 codes by TFBS Combination. Top 15 values, Fisher's Exact test used, expected values <5

6.3.2 Bootstrapping Report

The bootstrapping approach was applied to all combinations observed in the chi-squared and Fisher's exact tests, two runs of 1,000 randomly selected samples were initially created.

To create p-values for results achieved by bootstrapping an empirical value (s) has to be created where s is the count of the number of times the observed count has been exceeded or matched in the bootstrapped samples (Grimshaw et al. 1995). A p-value can then be created by:

$$p - value = \frac{\int s \geq observed}{N}$$

As this calculation is not as conservative as those created with Bonferroni adjustments, 52 combinations of ICD-10 codes and TFBS combinations were observed with two-tailed significance levels of < 0.025 or > 0.975 from the 1,000 iteration bootstrapping. Larger bootstrapping samples where then produced for the best performing results from both initial tests. The top ten results from both the chi-squared and Fisher's Exact tests were then analysed to further verify results.

The results of this expanded analysis are presented in table 6-5. Four combinations of TFBSs were not seen at all in the 25,000 samples (p-value = 0). At the other end of the scale, those where observed were less than expected, three combinations had highly significant p-values of < 0.01 (1 – 0.99). Eighteen of the 20 tests were significant at the 0.025 level due to the less exacting

significant requirements when removing multiple test restrictions.

TFBS Combination	ICD10	Description	25k	Emp p-val
E2F1,MAX,NFYA,SRF	H40	Optic disc parameters	0	0
CTCF,NFKB1	I26	Venous thromboembolism	0	0
NR1H2::RXRA	D59	Haemolytic anaemia in hep	0	0
AP1,CTCF,MAX,MYC::MAX,SP1,SPI1, TFAP2A,USF1,YY1	I45	Electrocardiographic traits	0	0
NFYA	Q23	Aortic root size	1	4.00E-05
MAX	C44	Cutaneous basal cell carcinoma	2	8.00E-05
SP1,USF1	I48	Atrial fibrillation	3	0.0001
E2F1,SP1,SPI1	B24	HIV-1 disease progression	6	0.0002
E2F1,MAX,NFYA,SPI1,USF1	C16	Diffuse-type gastric cancer	16	0.0006
CTCF,E2F1,MAX,SP1	C16	Diffuse-type gastric cancer	27	0.0011
USF1	G35	Multiple sclerosis	34	0.0014
NFYA,TFAP2A	K50	Crohns disease	121	0.0048
NFYA	I25	Coronary heart disease	398	0.0159
NFYA	G12	Sporadic Amyotrophic Lateral	604	0.0242
CTCF	Q35	Nonsyndromic cleft lip	959	0.0384
SPI1	M32	Systemic Lupus Erythematosus	1075	0.0430

Table 6-5 Empirical p-values of top scoring chi-squared and Fisher's Exact TFBS combination against ICD-10 codes. 25,000 Bootstrap samples. Observed > Expected

TFBS Combination	ICD10	Description	25k	Emp p-val	1 – Emp p-val
CTCF	J44	Spirometric measures of lung	24703	0.9881	0.0119
NFYA	M45	Ankylosing spondylitis	24777	0.9911	0.0089
CTCF,E2F1	G30	Alzheimer's disease	24853	0.9941	0.0059
CTCF	C43	Melanoma	24899	0.9960	0.0040

Table 6-6 Empirical p-values of top scoring chi-squared and Fisher's Exact TFBS Combination against ICD-10 codes. 25,000 Bootstrap samples. Expected > Observed

6.3.3 Number of TFBSs Repeats

The results presented so far are based solely on the presence or absence of TFBSs in combinations without examining, either the number of times they occur, their positions relative to the TSS and/or the order in which they appear. In order to examine these elements, the genes from the three most significant chi-squared test results were extracted for analysis along with the significant results from the Fisher's Exact tests. All six (USF1 Only, NFYA - TFAP2A, NFYA Only, E2F1 - MAX - NFYA - SRF, CTCF - NFKB1, NR1H2::RXRA) of these being confirmed as significant in the bootstrapping exercise (Table 6.5). The six TFBS combinations consisted of one with four distinct TFBSs, two with two distinct TFBSs and the other three just having the presence of a single TFBS. All of these TFBSs though, could have been observed more than once per gene.

The number of repeats of each TFBS within a TFBS combination is shown in figure 6-1. For the Fisher's Exact test results the numbers were small with either a single TFBS, two, or three repeats being seen for each TFBS. The numbers for

the chi-squared results were higher, with the most commonly observed number of repeats being one, the highest being the TFAP2A TFBS in the NFYA,TFAP2A TFBS combination where 15 repeats were observed.

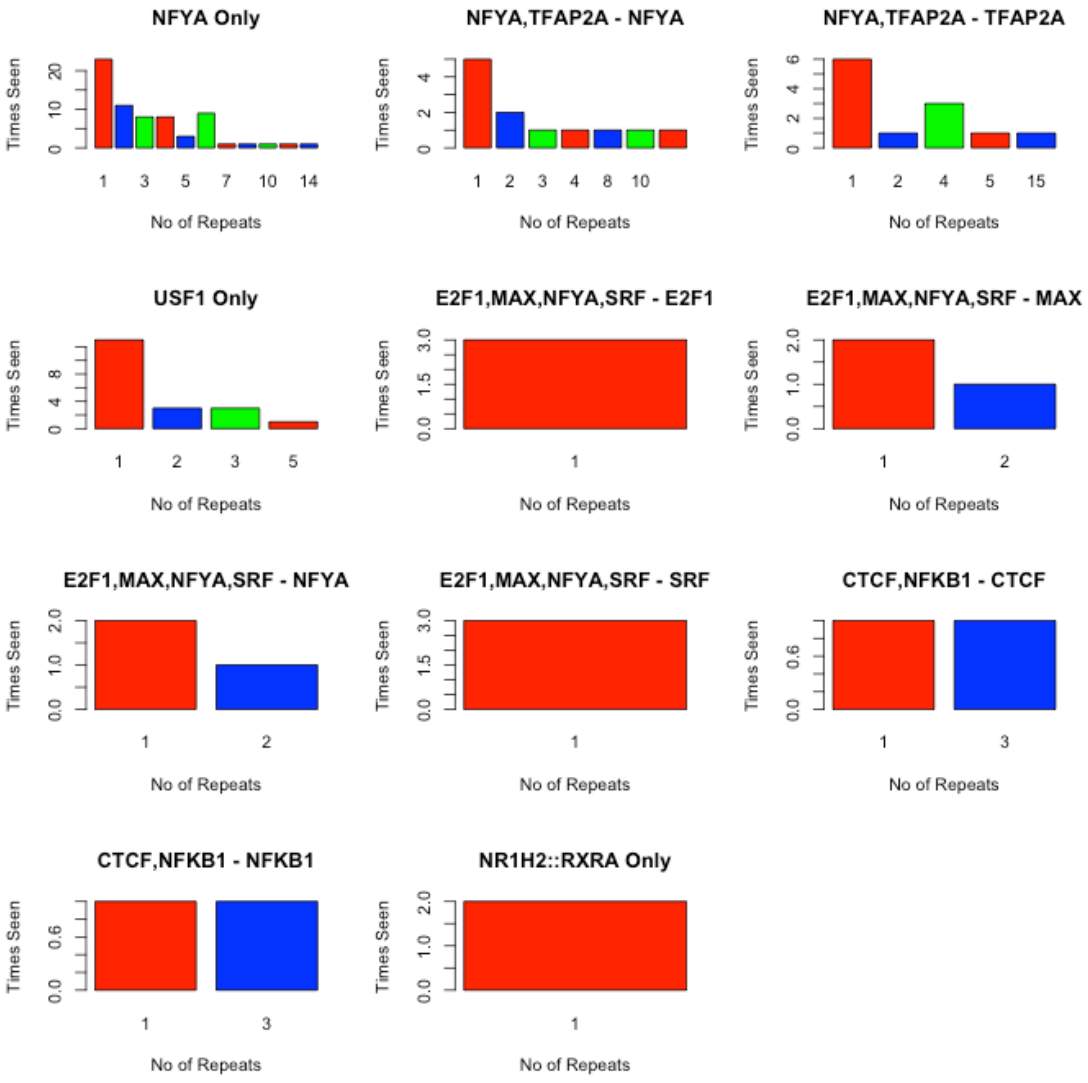

Figure 6-1 No of TFBS repeats within TFBS Combination. X-axis bars represent the number of TFBS repeats. Y-axis represents the number of times observed.

6.3.4 Examination of TFBS Position

To examine the order and position of individual TFBSs, the results for the NFYA Only, NFYA – TFAP2A, and USF Only combinations have been analysed in detail. These three examples were selected as they had 67, 20, and 12 observations respectively and were significant in the bootstrapping analysis. Figure 6-2 shows the offset position of each NFYA TFBS in each of the 67 genes where it was observed; figure 6-3 shows both NFYA and TFAP2A for the 12 genes where they appeared in combination and figure 6-4 shows USF1 Only. The striking pattern of large numbers of TFBSs being seen between 1100bp and 1300bp was investigated by producing a more detailed view in figure 6-5 although no further distinct patterns were observed. Additionally, those genes with two or more TFBS downstream of the TSS are presented in more detail in figure 6-6.

In addition, counts and mean offsets are shown for TFBSs within the six key combinations (USF1 Only, NFYA - TFAP2A, NFYA Only, E2F1 - MAX - NFYA - SRF, CTCF - NFKB1, NR1H2::RXRA) in table 6-7. The observed counts and means are also presented for overall false positives (obtained from calculating JASPAR PWMs that were not verified by ENCODE (Thomas et al. 2007)), overall true positives (JASPAR PWMs verified by ENCODE), and the full ENCODE data. In those genes with just NFYA TFBSs present, the average offset observed was 1006bp upstream of the TSS, which is markedly different to the 655bp average offset seen for the false positives. With true positives NFYA TFBSs, a much closer result of 945bp upstream is observed, and the figure for all those found on the ENCODE dataset, a mean of 950bp upstream, is closer still (See Table 6-7).

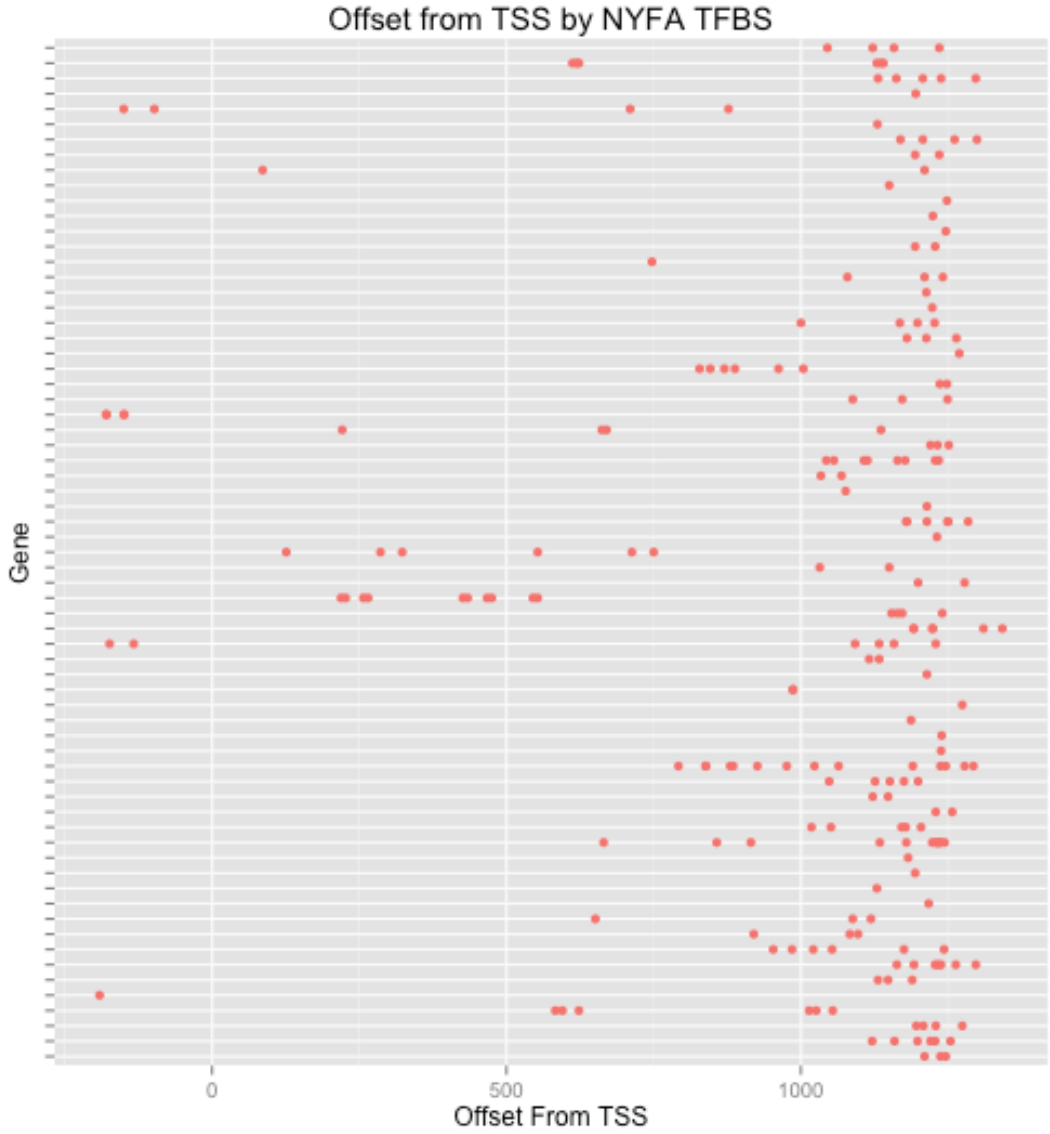

Figure 6-2 Offset from TSS for the NFYA TFBS in the NFYA Only TFBS combination.

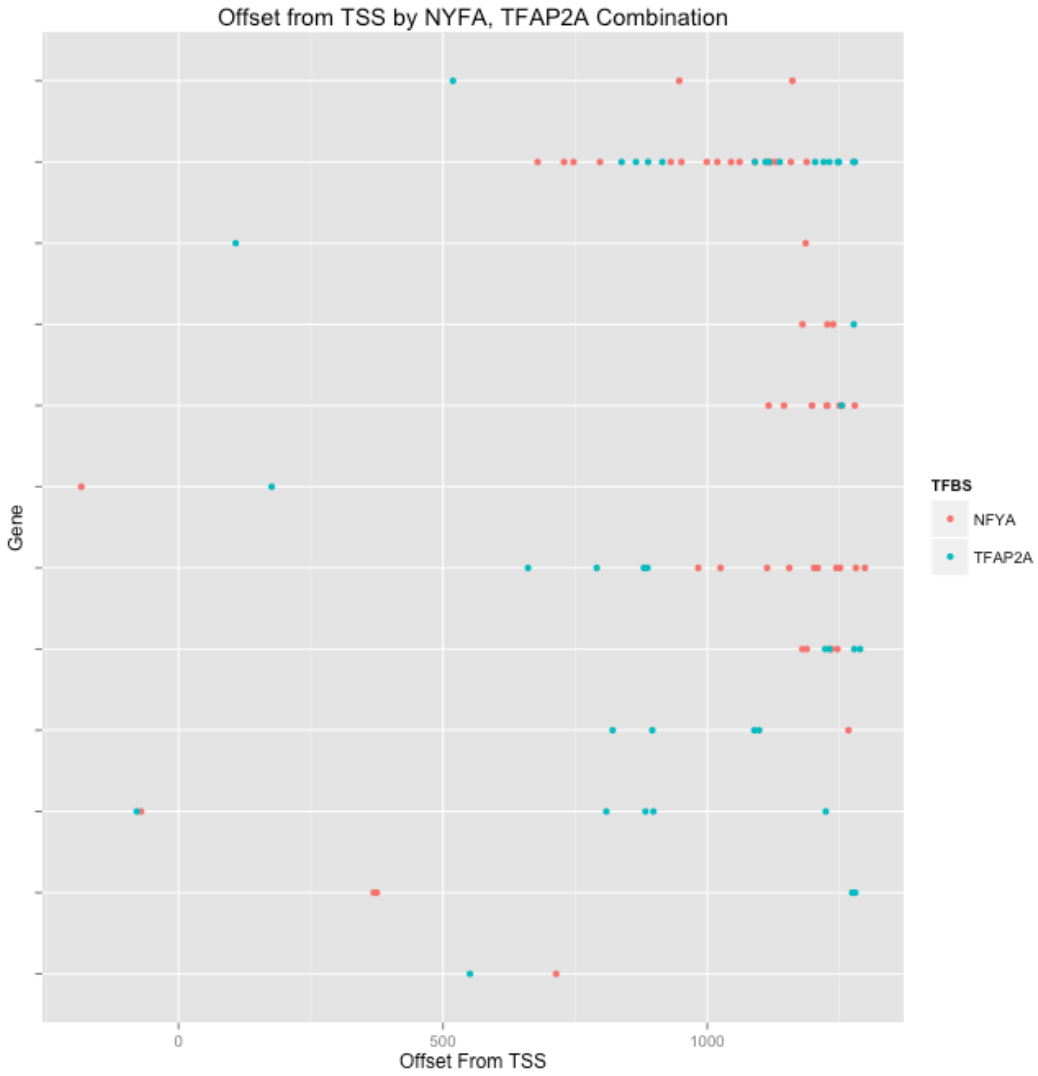

Figure 6-3 Offset from TSS for the NFYA, TFAP2A TFBS in combination.

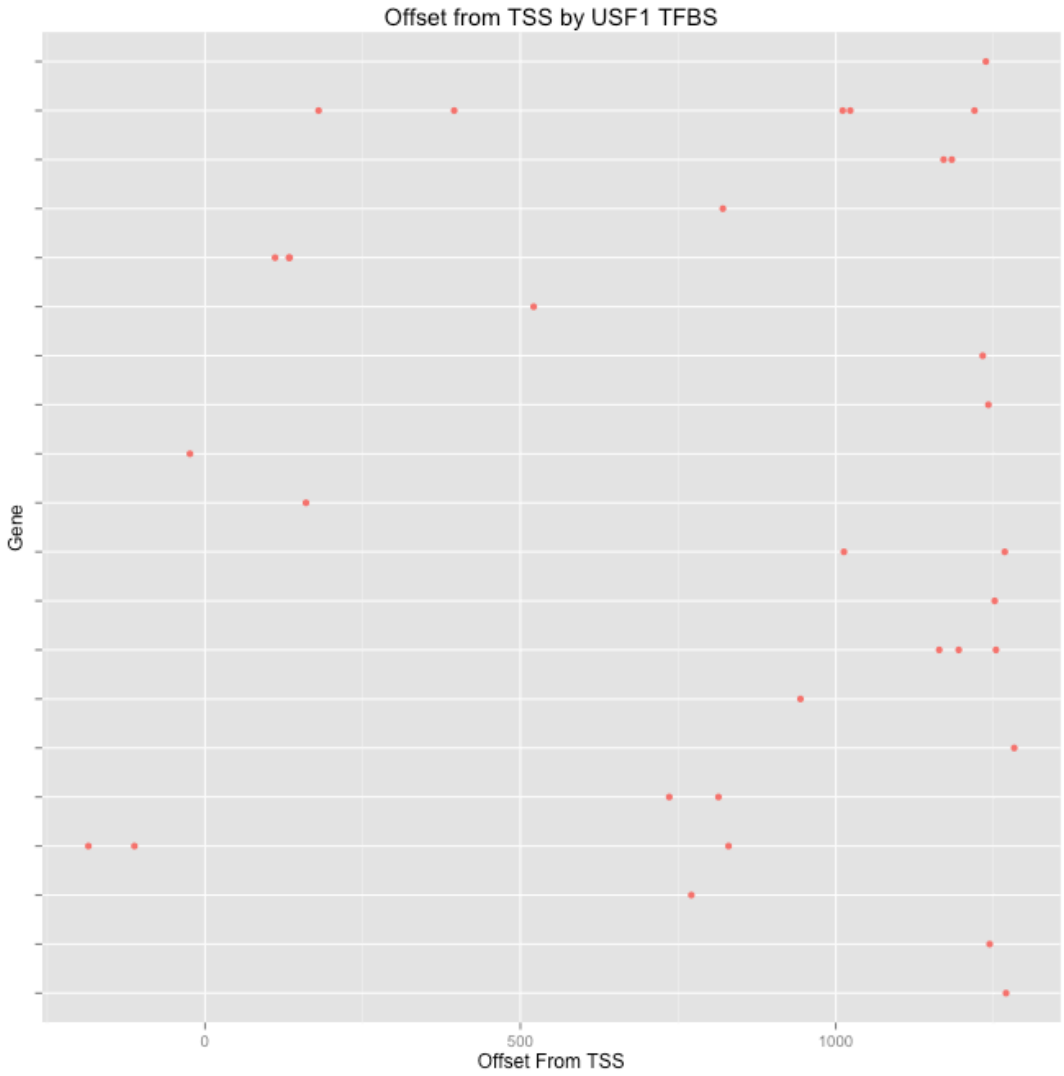

Figure 6-4 Offset from TSS for the USF Only Genes.

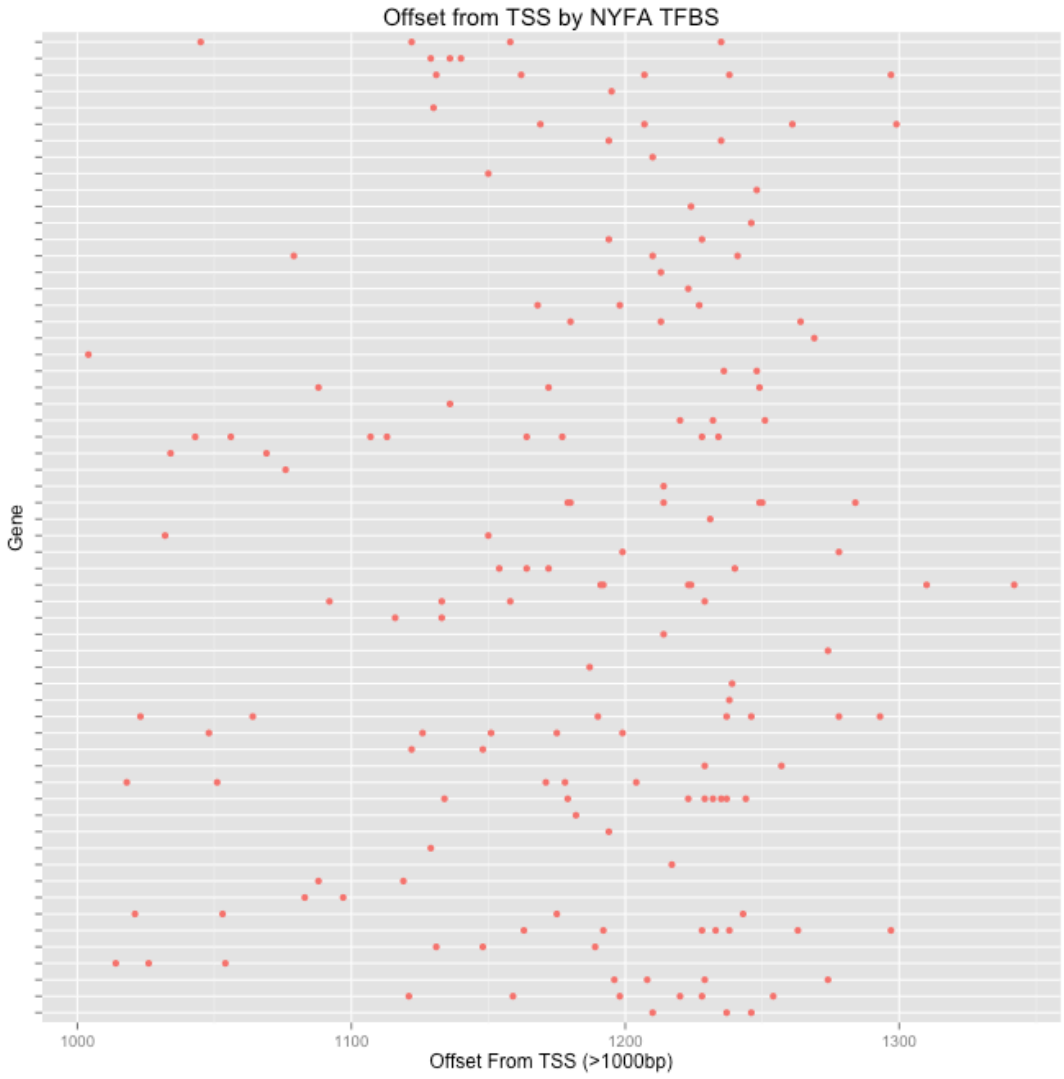

Figure 6-5 Drill down of NFYA TFBS in the NFYA Only TFBS combination.

Genes showing Offset positions of TFBS > 1000 bp from TSS.

TFBS Comb.	TFBS	Cnt	MO	FP Cnt	FP MO	TP Cnt	TP MO	ENC Cnt	ENC MO
CTCF,NFKB1	CTCF	4	989	38838	714	6405	688	12894	694
CTCF,NFKB1	NFKB1	4	341	74118	672	516	764	406	686
E2F1,MAX,NFYA,SRF	E2F1	3	1047	84889	734	2224	852	3367	905
E2F1,MAX,NFYA,SRF	MAX	4	699	173302	621	3668	845	8593	872
E2F1,MAX,NFYA,SRF	NFYA	4	285	82198	655	5291	945	9069	950
E2F1,MAX,NFYA,SRF	SRF	3	967	13903	615	190	841	515	812
NFYA	NFYA	220	1007	82198	655	5291	945	9069	950
NFYA,TFAP2A	NFYA	50	1029	82198	655	5291	945	9069	950
NFYA,TFAP2A	TFAP2A	40	975	1293988	707	9846	833	8468	798
NR1H2::RXRA	NR1H2::RXRA	2	1103	1106	658	37	891	214	742

Table 6-7 Counts (Cnt) and Mean Offsets (MO) for TFBS combinations for observed, all false positives (FP), all true positives (TP), and all ENCODE (ENC).

6.4 Discussion

The application of the TFBS predictions to the analysis of genes within LD blocks, associated with specific disease conditions or traits, revealed its potential to provide insight into transcription regulation. The analysis of the combination and positioning of those combinations of TFs was restricted by very small numbers of observations for most LD block genes sets. In addition, when the

offset positions for the NFYA TFBS are examined for the 67 genes in a LD block linked to coronary heart disease (Figure 6-3), the frequently occurring nature of the TFBSs due to the small average size of their motifs make it impossible to extract potentially conserved repeat patterns between subsets of genes by eye. A method is required to extract this information, an area discussed in Chapter 8.

The combinations and position of the NFYA, TFAP2A and USF1 TFBSs were analysed in detail. In the current data, NFYA was overrepresented in genes from a LD block associated with coronary heart disease; E2F1 was associated with optic disc conditions and USF1 with multiple sclerosis (see Table 6-3 and 6-4). In support of these observations, a number of experimental studies have been published implicating these specific TFs in these disease conditions. NFYA transcription factor has been linked to the HSPA1A gene that produces heat shock (or stress) proteins and has been associated with increased susceptibility to coronary heart disease (Sasi et al. 2013). E2F1 overexpression, a TF associated with apoptosis and cell proliferation, has been observed in retinal degeneration in optic discs in mice (Chen & Nathans 2007) and USF1 (along with other TFs) expression levels have been connected to multiple sclerosis. In the last case, TFs have been linked to expression of the major histocompatibility complex (MHC) in neurodegenerative disease (Gobin et al. 2001)

Although it has been commonplace to look for regulatory effects, positive and negative, within 1000bp of the TSS e.g. (Cooper et al. 2006; Hannenhalli 2008; Veerla et al. 2010), 61% of the NFYA TFBSs were seen between 1100-1300 bp upstream. Together with the entropy profiles (See 2.4.3), these results suggest that the search for TFBSs should include promoter regions up to at least 1300bp

upstream, as using a 1000bp threshold means that many functional TFBSs will be missed.

Whilst the analysis of TFBSs combinations was limited by the data that is currently available, the GWAS catalog (Hindorff et al. 2011) is continually growing and further large scale projects such as ENCODE (Thomas et al. 2007) will provide data on additional TFBS in the future. Hence, the current method has the potential to reveal new CRMs when sufficient data becomes available and a method for extracting conserved patterns in genes with a high number of TFBS repeats has been developed.

This chapter looked at the application of the model and ENCODE experimentally verified TFBSs to the universe of publically available GWAS. The results took into account the linkage disequilibrium blocks of the genes containing the TFBSs and examined the statistical validity of over represented combinations potentially acting in *cis*-regulatory modules.

The next chapter details collaborations that have taken place during the production of this thesis.

7 Role of Transcription Factors in Infection and Immunity and Contributions to Other Studies

During the course of my PhD I contributed to additional research projects that resulted in a number of publications (published or under review) that are summarised in this chapter. The first two papers focus on transcription factors, firstly in the EBV genome and secondly their combinatorial role in interferon-gamma responses in Humans. Three more papers are then presented detailing research into T-cells and their role in immunity. The title and authors of a fourth paper being reviewed are also recorded, but due to intellectual property issues, the data within the paper cannot be detailed prior to publication.

7.1 Epigenetic Control of viral life-cycle by a DNA-methylation dependent transcription factor

Authors: Kirsty Flower, David Thomas, James Heather, Sharada Ramasubramanyan, Susan Jones, Alison Sinclair

Journal PloS one

Date 11/10/2011

Volume 6

Issue 10

Pages e25922

7.1.1 Overview

The Epstein-Barr virus (EBV) is a virus of the herpes family and is extremely common in humans. The EBV genome has a biphasic cycle, after lytic replication it resets to an unmethylated state, becoming more methylated during the latent phase. The transcription factor Zta interacts with Zta Response elements (ZREs) and it is expressed transiently following infection and again when the virus switches between the latent state and lytic replication. The requirement for CpG methylation at critical ZREs could regulate EBV replication. Specifically, immediately after infection, it could prevent replication in the non-methylated genome retaining latency and later aid the activation of lytic genes as the genome becomes more methylated.

A new computational approach was used to predict the location of ZREs and determine which ones were CpG methylation dependent with the results verified using in vitro and in vivo assays. Results showed that the majority of lytic cycle genes have at least one, and many have multiple, copies of methylation-dependent CpG ZREs within their promoters. This suggests that the methylation status of the EBV genome together with the amount of Zta act in parallel to control the expression of lytic genes.

7.1.2 Personal Contribution

I performed all of the in silico work for this project. After extracting and loading the EBV genome into a local database, code was written to scan rolling windows of DNA for sequence regions matching position weight matrices (PWMs) of ZREs. These matches were provided to researchers for experimental verification using electrophoretic mobility shift assays (EMSAs). Database tables of the location and sequence of the predicted ZREs were built to extract experimental data and to produce the figures for the paper.

The database containing the EBV genome was also used as a back-end for a web application that I built in PHP allowing EBV researchers to:

- Extract Upstream DNA Sequences
- Extract Genetic DNA Sequences
- See ZREs within 1000bp of a gene
- Calculate number of ZREs upstream within a specific number of base pairs upstream of a gene start
- View closet ZRE to each gene start
- View closest ZRE to each gene start (by class)

(See http://bioinf.biochem.sussex.ac.uk/EBV)

7.2 Identification of interferon-gamma response genes: from genetic linkage peaks to transcription factor networks

Under Review by The Journal of Genetics and Genomics (Submitted November 2013)

Authors: Elizabeth Hellen, Melanie Newport, David Thomas, Chris Finan, Susan Jones

7.2.1 Overview

The identification of genes involved in complex traits can be achieved via linkage peak mapping or GWAS but both have limitations. Genetic linkage can extend across large genomic distances leading to identification of broad chromosomal regions comprising hundreds of genes. GWASs can identify multiple significant makers that map in or near multiple genes. Hence, both approaches result in gene sets in which only a small number of genes (classed as true positive (TP) genes) have a functional role in the disease with the remainder being false positives (FP) genes. The problem is to differentiate the TP from the FP genes. Here, a method was developed for the identification of TP genes based on the hypothesis that genes sharing specific transcription factors (TFs) have related functions.

This method, a naïve bayes classifier (NBC) was applied to a dataset of 392 genes in significant linkage peaks from a study to assess interferon-γ response to Mycobacterial antigens in humans. The NBC, that combined data from six TFBSs

in gene promoters, produced a cross validated result of 80.5% accuracy. The predicted TFs from the NBC were analysed by creating a network graph in which the nodes were genes and the edges shared transcription factors. This analysis revealed 18 genes sharing five TFBSs. Of these, three had TFBSs of AP1 and GATA1 conserved in terms of order and position within the promoter. This combination potentially represents a CRM for the regulation of interferon-γ responses in Mycobacterial infection.

7.2.2 Personal Contribution

I performed the verification of the results using ChIP-Seq data from ENCODE (Birney et al. 2007). The candidate genes and their up and downstream sequences were extracted from the GRCh37.6 human genome assembly. In addition, ChIP-Seq verified TFBS for the relevant TFs (AP1, GATA1, GATA2, USF) shared by the candidate genes, was extracted from the Ensembl Regulatory build v66 from the same human genome assembly. Reports were then produced to compare NBC and network graph predictions with experimentally verified TFBSs.

7.3 Immunological Data Pipeline and Results Database

The remaining publications in this chapter utilise biological data obtained using flow cytometry. This technique uses laser light for counting and identification of microscopic particles, including cells, chromosomes, and biomarkers (Chattopadhyay & Roederer 2012). The work presented in the next four papers has seen the integration and analysis of data from the polychromatic flow cytometry where initial data is recorded using the FlowJo (http://www.flowjo.com/) software. I have written software to take the raw flow cytometry data and process it in a flexible computational pipeline to produce required reports and aid further analysis.

A summary of the typical pipeline steps would be:

- Data is retrieved from FlowJo for individuals consisting of a number of tubes (typically 20) with cell counts and florescence intensities related to various stimulations. Typically five stimulations are handled in parallel via binary gating resulting in 559 measurements per tube. This input data is inserted into database tables.

- A control file spreadsheet is used to specify which actions should be performed on which data item. For example, which tube is the control tube, which data should be divided by the total of CD4 cells, which need to be classified in various groupings etc.

- A java application is run that reads the input data and the control file and performs the recalculation, reformatting, subtotalling, and subtraction of control tubes. This populates new tables in the database and provides summary reports for the researcher.

- A data quality program is then run to categorise the quality of the data in terms of reliability.

- Patient pathology data is held anonymously is further tables and this is merged as required to aggregate and differentiate groups for analysis.

- A flattening process is available to extract key fields from the various tubes and present them as a single record that can be more easily analysed in statistical packages.

The end result of the pipeline is data that can be selected via database queries or custom programs to create reports or a variety of files for further analysis in systems such as Excel (Microsoft, USA) or PASW (IBM, USA). This system has been used in the data processing for the next four papers.

7.3.1 The phenotypic distribution and function profile of tuberculin-specific CD4 T-cells characterizes different stages of TB infection

Authors: Mathias Streitz, Stephan Fuhrmann, David Thomas, Elizabeth Cheek, Laurel Nomura, Holden Maecker, Peter Martus, Nima Aghaeepour, Ryan R Brinkman, Hans-Dieter Volk, Florian Kern

Journal Cytometry Part B: Clinical Cytometry

Date 1/11/2012

Volume 82

Issue 6

Pages 360-368

Received Best Original Paper Award 2012-2013 by Clinical Cytometry

Primary infection with Mycobacterium tuberculosis normally results in latent tuberculosis infection (LTBI) that has been estimated to affect one-third of the world's population. LTBI converts to pulmonary or extra-pulmonary TB at a lifetime rate of 10% in otherwise healthy people. The best test for detecting TB infection are interferon-γ release assays however they cannot distinguish between active TB and LTBI.

Comparisons were performed on samples taken from ex-vivo tuberculin activated CD4 T-cells from three groups, those with active pulmonary TB, long term exposed hospital staff (some with LTBI), and unexposed university staff.

The selected activation markers examined were CD154 upregulation, IFN-γ, TNF-α, IL-2, and degranulation.

The best distinguishing combination, in terms of identifying active TB against LTBI, from the 32 (2^5) combinations was CD154+, TNF-α+, IFN-γ -, IL2-, degranulation- with an area under the ROC curve of 0.90. However an effective (easier, cheaper) alternative was the ratio of TNF-α+/ IFN-γ+ CD4 T-cells that produces an area under the ROC curve of 0.87.

7.3.2 A novel CMV-induced regulatory type T-cell subset increases in older life and links virus-specific immunity to vascular pathology

Authors: Nadia Terrazzini, Martha Bajwa, Serena Vita, Elizabeth Cheek, David Thomas, Nabila Seddiki, Helen Smith, Florian Kern

Journal Journal of Infectious Diseases

Date 7/11/2013

Pages jit576

Cytomegavirus (CMV) directly targets vascular muscles both endothelium and smooth. At older ages it is associated with accelerated vascular disease and hence mortality.

Alongside conventional ex vivo activation-induced T-cell responses to CMV antigens, CMV-induced regulatory-type CD4 T-cells (iTregs) were measured in a novel protocol. Donors comprised 131 healthy 60-85 year olds and results were compared to a sample of 55 healthy younger people of between 20 and 35 years old.

Results showed that frequencies of iTregs and CMV-specific CD8 T-cells were significantly associated with diastolic and mean arterial pressures even when taking into account confounders such as age, BMI and smoking. Whilst CD8 T-cell might cause vascular problems, iTregs may attenuate this response.

7.3.3 Cytomegalovirus infection modulates the phenotype and functional profile of the T-cell immune response to mycobacterial antigens in older life

Authors: Nadia Terrazzini, Martha Bajwa, Serena Vita, David Thomas, Helen Smith, Rosanna Vescovini, Palo Sansoni, Florian Kern

Journal Experimental Gerontology

Date Accepted 18/12/13

Cytomegalovirus (CMV) infection is associated with accelerated decline in the immune system due to age, immunosenescence.

Most people aged 60 and above have specific immunity to Mycobacterial tuberculosis due to vaccination, exposure or both. When response to tuberculin was compared to a control group of younger people (20-35) no significant differences were observed although an increase in outliers was apparent in the older group. A comparison of tuberculin response and CMV T-cell response however showed significant correlation between younger and older people. Furthermore, the likelihood of tuberculin-induced T-cells becoming terminally differentiated varied proportionally to the size of CMV T-cell response.

These results show that CMV serostatus has a fundamental impact on immune response to mycobacterial antigens in later life.

7.3.4 Analysis of CMV induced T cell memory inflation reveals response complexity, diversity, and breadth in humans

Authors: Andres Sylwester, Kate Nambiar, Serena Vita, Marhta Bajwa, Nadia Terrazzine, Stefano Caserta, Helen Smith, Elizabeth Cheek, David Thomas, Paul Klenerman, Louise Picker, Florian Kern

Under Review (Submitted July 2013)

Collaborators do not want to release details of this paper before publication.

8 Discussion

8.1 Summary of novel methods and results

This thesis has presented original work using data from the entire human genome assembly (Venter et al. 2001) in two main areas.

(a) The development of machine learning models to predict functional TFBSs based on parameters derived from the sequence and structure of promoter DNA including DNA entropy.
(b) The application of the models (together with experimentally verified TFBSs) to phenotypic data derived from GWASs with the aim of identifying CRMs.

These areas directly align with the aims of the research presented in Chapter 1. The initial work to include DNA entropy into the TFBS prediction models revealed two very interesting results that have not been observed in previous work. The first was that functional TFBSs have a higher entropy (or information content) than non-functional sites. The second was that entropy profiles enabled the promoters of genes from different broad classes (constitutive and facultative) to be differentiated. In addition, the work on the application of the models to the prediction of TFBSs in gene sets from LD blocks revealed that TFBSs are clustered in a region around 1200bp upstream of the TSS, 61% of NFYA TFBSs, in genes from the LD block linked to coronary heart disease, are found within 100bp of 1200bp upstream.

8.2 Limitations

Whilst the results outlined above are exciting developments, any work based on large volumes of genomic data has significant limitations that have been recognised. The main problems faced during this work were the availability of experimental data on TFBS and of databases of potential or experimentally verified CRMs.

The key feature of the final work was the integration of TFBS data from the human genome via the ENCODE project (Thomas et al. 2007). At the start of the thesis only the draft ENCODE data was available, that only covered 1% of the human genome (Birney et al. 2007). Initial models were constructed using the draft data, but as the complete ENCODE data (Bernstein et al. 2012)became available during the course of the thesis these initial models were abandoned and new ones tested on the complete data. This significant change in data availability resulted in a major amount of time being committed to the modelling stage of the work and led to the final modelling being tested using a reduced number of techniques. This is the reason that techniques such as SVMs are not included within the thesis.

The data available within the most recent ENCODE update should be viewed as giving an incomplete picture of TF binding in gene promoters. The JASPAR database (Sandelin et al. 2004) at the time of extraction (Jan 2010) had 80 human TFBS PWMs. Using ENCODE (Bernstein et al. 2012)for experimental verification however resulted in only 18 of these being directly comparable. Although a valid assumption is that the ENCODE research teams worked on the most important TFs, a bias has potentially entered the models by using only

~25% of the original JASPAR motifs. The ChIP-Seq method is both time consuming and costly and is limited by the availability of antibodies. Hence, its use to cover promoters of the complete human genome is still limited, and many 10s of thousand TFBS still remain to be identified experimentally.

The aim at the start of this work was to compare the potential CRMs identified with those already stored in CRM databases, and to use data from these databases to validate the results on a wider scale. However, resources that were planned for use, such as ORegAnno (Griffith et al. 2008)and cisRED (Robertson et al. 2006)went from being active at the start of the project to "cobweb" sites, with no updates since 2009 for cisRED and 2008 for ORegAnno. This common problem, often down to grants running out or key staff moving on, led to the use of these sites being abandoned.

8.3 Recent Developments

Enhancements in data gathering techniques are improving and the amount of data available is continually increasing. ENCODE (Becker 2011) are involved in on-going projects to enhance the data they have available, improved analysis techniques and provide new experimental data. This has improved their tracks, their annotation layers that are matched to genetic coordinates, in USCS and Ensembl browsers at the end of 2013 (Flicek et al. 2013; Karolchik et al. 2013).

In the current work, conclusions regarding potential novel CRMs were limited by

the data available from GWASs. However, the number of published GWASs is increasing dramatically with the repository at the NIH adding approximately 500 studies in 2012 alone (Hindorff et al. 2011) with that number set to increase substantially in the coming years.

The need for better methods to identify functional TFBSs to increase knowledge of transcriptional control of gene expression in different systems means that new methods are continually being published. One method, which features an updated form of PWMs that allows variable length motifs and position independence, has been published very recently (Mathelier & Wasserman 2013). This work introduced the concept of Transcription Factor Flexible Models (TFFM) that are based on a HMM produced by analysing the ENCODE ChIP-Seq data. Most techniques used for examining a DNA sequence for the prediction of TFBSs assume that the motif will be the same size, however, in many cases there is flexibility in the length and positional arrangement of the bases (Tomovic & Oakeley 2007). TFFM models provide a framework that can handle variable flanking regions and position dependencies allowing potentially much better accuracy and reducing the number of PWM false positives. The development of such as model has only been possible because of the increased data available through the ENCODE project.

8.4 Future Work

The research in this thesis is obviously time restricted, and there are a number of key areas where further work is possible.

Modelling software: Modelling software is being continually updated; the chosen Encog framework (Heaton 2010) now offers various different types of models that could be tested including SVMs and Bayesian Networks. The number of packages performing every conceivable of types of analysis in R (Team 2005) is growing every month, and with automatic links into the MySQL based SETS database, could be simply run from the R platform.

Additional data for application: In addition to enhancing the modelling by including newly published GWASs, different groups of functional similar genes could be analysed by using gene ontologies. A further approach would be to consider phylogenetic data and apply a conservation filter to predicted TFBSs; although this would restrict the identification of novel TFBSs that might be species specific.

New methods of TFBS pattern analysis: The work in chapter 6 applied the TFBS prediction method as far as presenting data on the offset position of multiple TFBS for sets of genes within 3 LD blocks (Figure 8-1). However, the data presented for NFYA TFBSs in 67 genes and the number of repeats made it impossible to identify conserved patterns between genes easily by eye. For example, a conserved pattern representing a CRM might comprise three TFBS repeats a conserved distance apart, but with small variations in absolute offset and distance being permitted (Figure 8-1). A significant area of further work is

to develop a system for unsupervised searching for such matches of repeats within multiple gene promoters, potentially using fuzzy inference systems.

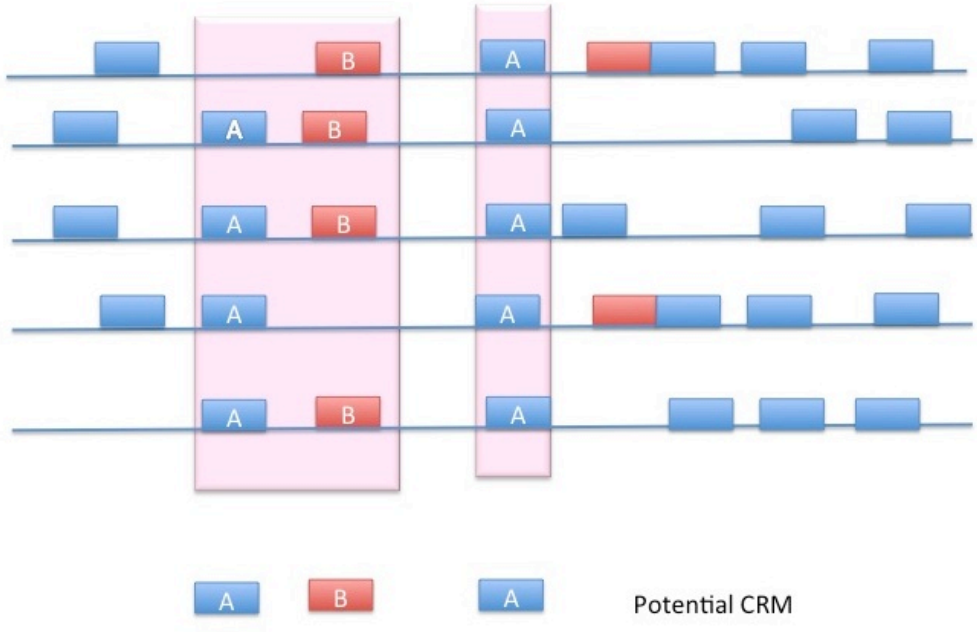

Figure 8-1 Potential cis-regulatory module showing TFBSs (A,B,C) in a conserved pattern over several genes.

8.5 Conclusion

The work in this thesis presents a new model for TFBS prediction, offers insights into the positioning of TFBSs within gene promoters, and variation in the entropy landscape of constitutive and facultative gene promoters. The work has been conducted over four years during which time both genomic data and analysis methods (both experimental and computational) have increased at a rate not previously seen. As the cost of sequencing decreases and as quality increases, even more data will become available. This will enable the application of TFBS prediction models (such as those described in this thesis, and ones developed in the future) to even larger datasets to give further insight into the complex mechanism of transcription regulation.

Reference List

Adler, R.L., 1979. Topological entropy and equivalence of dynamical systems. *American Mathematical Soc*, 219.

Allocco, D.J., Kohane, I.S. & Butte, A.J., 2004. Quantifying the relationship between co-expression, co-regulation and gene function. *BMC Bioinformatics*, 5, p.-.

Annunziato, A., 2008. DNA Packaging: Nucleosomes and Chromatin. *Nature Education*.

Ashburner, M. et al., 2000. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, 25(1), pp.25–29.

Barski, A. & Zhao, K., 2009. Genomic Location Analysis by ChIP-Seq. , 107(1), pp.11–18.

Basheer, I.. & Hajmeer, M., 2000. Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43(1), pp.3–31.

Becker, P.B. ed., 2011. A User's Guide to the Encyclopedia of DNA Elements (ENCODE). *PLoS Biology*, 9(4), p.e1001046.

Benson, D. a et al., 2006. GenBank. *Nucleic acids research*, 34(Database issue), pp.D16–20.

Bernstein, B.E. et al., 2012. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414), pp.57–74.

Bickhart, D.M. & Liu, G.E., 2013. Identification of Candidate Transcription Factor Binding Sites in the Cattle Genome. *Genomics, proteomics & bioinformatics*, 11(3), pp.195–198.

Bickmore, W. a, 2013. The Spatial Organization of the Human Genome. *Annual review of genomics and human genetics*, (July), pp.1–18.

Birney, E. et al., 2007. Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. *Nature*, 447(7146), pp.799–816.

Bishop, C., 1995. *Neural Networks for Pattern Recognition*,

Brazma, a et al., 2001. Minimum information about a microarray experiment (MIAME)-toward standards for microarray data. *Nature genetics*, 29(4), pp.365–71.

Brown, C.D., Johnson, D.S. & Sidow, A., 2007. Functional architecture and evolution of transcriptional elements that drive gene coexpression. *Science*, 317(5844), pp.1557–1560.

Butler, J.E.F. & Kadonaga, J.T., 2002. The RNA polymerase II core promoter: a key component in the regulation of gene expression. *Genes & development*, 16(20), pp.2583–92.

Cabin, R.J. & Mitchell, R.J., 2000. To Bonferroni or not to Bonferroni: when and how are the questions. *Bulletin of the Ecological Society of America*, 81(3), pp.246–248.

Cairns, B.R., 2009. The logic of chromatin architecture and remodelling at promoters. *Nature*, 461(7261), pp.193–8.

Chang, C.-W. et al., 2011. Identification of human housekeeping genes and tissue-selective genes by microarray meta-analysis. *PloS one*, 6(7), p.e22859.

Chattopadhyay, P.K. & Roederer, M., 2012. Cytometry: today's technology and tomorrow's horizons. *Methods (San Diego, Calif.)*, 57(3), pp.251–8.

Chen, J. & Nathans, J., 2007. Genetic ablation of cone photoreceptors eliminates retinal folds in the retinal degeneration 7 (rd7) mouse. *Investigative ophthalmology & visual science*, 48(6), pp.2799–805.

Colosimo, a & De Luca, a, 2000. Special factors in biological strings. *Journal of theoretical biology*, 204(1), pp.29–46.

Cooper, S. et al., 2006. Comprehensive analysis of transcriptional promoter structure and function in 1% of the human genome. *Genome*, pp.1–10.

Crooks, G.E. et al., 2004. WebLogo : A Sequence Logo Generator. , pp.1188–1190.

Daenen, F., van Roy, F. & De Bleser, P.J., 2008. Low nucleosome occupancy is encoded around functional human transcription factor binding sites. *BMC Genomics*, 9, p.-.

Dunham, I. et al., 2012. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414), pp.57–74.

Eddy, S., 1998. Profile hidden Markov models. *Bioinformatics*, 14(9), pp.755–763.

Elgin, S.C.R., 1988. The Formation and Function of Dnase-I Hypersensitive Sites in the Process of Gene Activation. *Journal of Biological Chemistry*, 263(36), pp.19259–19262.

Fahlman, S.E., 1988. An empirical study of learning speed in back- propagation networks.

Flicek, P. et al., 2012. Ensembl 2012. *Nucleic acids research*, 40(Database issue), pp.D84–90.

Flicek, P. et al., 2013. Ensembl 2014. *Nucleic acids research*, (8), pp.1–7.

Flicek, P. et al., 2010. Ensembl's 10th year. *Nucleic acids research*, 38(Database issue), pp.D557–62.

Gardiner-Garden, M. & Frommer, M., 1987. CpG islands in vertebrate genomes. *J Mol Biol*, 196(2), pp.261–282.

Gobin, S.A.M.J.P., Montagne, L. & Zutphen, M.V.A.N., 2001. Upregulation of Transcription Factors Controlling MHC Expression in. , 77(February), pp.68–77.

Graur, D. et al., 2013. On the immortality of television sets: "function" in the human genome according to the evolution-free gospel of ENCODE. *Genome biology and evolution*, 5(3), pp.578–90.

Griffith, O.L. et al., 2008. ORegAnno: an open-access community-driven resource for regulatory annotation. *Nucleic Acids Res*, 36(Database issue), pp.D107–13.

Grimshaw, S.D., Efron, B. & Tibshirani, R.J., 1995. An Introduction to the Bootstrap. *Technometrics*, 37(3), p.341.

Haddrill, P.R. et al., 2005. Patterns of intron sequence evolution in Drosophila are dependent upon length and GC content. *Genome biology*, 6(8), p.R67.

Hall, M., Frank, E. & Holmes, G., 2009. The WEKA data mining software: an update. *ACM SIGKDD …*, 11(1), pp.10–18.

Håndstad, T. et al., 2011. A ChIP-Seq benchmark shows that sequence conservation mainly improves detection of strong transcription factor binding sites. *PloS one*, 6(4), p.e18430.

Hanley, J., 1982. The Meaning and Use of the Area under a Receiver Operating (ROC) Curve Characteristic. , (4).

Hannenhalli, S., 2008. Eukaryotic transcription factor binding sites--modeling and integrative search methods. *Bioinformatics*, 24(11), pp.1325–1331.

HAWKS, J., 2005. A haplotype map of the human genome.

Heaton, J., 2010. *Programming Neural Networks with Encog 2 in Java*,

Hindorff, L. et al., 2011. A catalog of published genome-wide association studies.

Hirschhorn, J.N. & Daly, M.J., 2005. Genome-wide association studies for common diseases and complex traits. *Nature reviews. Genetics*, 6(2), pp.95–108.

Ho, J.W.K. et al., 2011. ChIP-chip versus ChIP-seq: lessons for experimental design and data analysis. *BMC genomics*, 12(1), p.134.

Holland, R.C.G. et al., 2008. BioJava: an open-source framework for bioinformatics. *Bioinformatics (Oxford, England)*, 24(18), pp.2096–7.

Hopfield, J.J., 1982. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. *Proceedings of the National Academy of Sciences*, 79(8), pp.2554–2558.

Hsu, C., Chang, C. & Lin, C., 2010. A Practical Guide to Support Vector Classification. *Bioinformatics*, 1(1), pp.1–16.

Huang, D.W., Sherman, B.T. & Lempicki, R.A., 2009. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nature Protocols*, 4(1), pp.44–57.

Jagannathan, V. et al., 2006. HTPSELEX - a database of high-throughput SELEX libraries for transcription factor binding sites. *Nucleic Acids Research*, 34, pp.D90–D94.

Jiang, C.Z. & Pugh, B.F., 2009. Nucleosome positioning and gene regulation: advances through genomics. *Nature Reviews Genetics*, 10(3), pp.161–172.

Jothi, R. et al., 2008. Genome-wide identification of in vivo protein-DNA binding sites from ChIP-Seq data. *Nucleic Acids Res*, 36(16), pp.5221–5231.

Kaelbling, L., Littman, M. & Moore, A., 1996. Reinforcement learning: A survey. *arXiv preprint cs/9605103*, 4, pp.237–285.

Karamanos, K. et al., 2006. Statistical compressibility analysis of DNA sequences by generalized entropy-like quantities: towards algorithmic laws for biology? In *Proceedings of the 6th WSEAS International Conference on Applied Informatics and Communications*. World Scientific and Engineering Academy and Society (WSEAS), pp. 481–491.

Karolchik, D. et al., 2013. The UCSC Genome Browser database: 2014 update. *Nucleic acids research*, pp.1–7.

Kohonen, T., 1990. The self-organizing map. *Proceedings of the IEEE*, 78(9), pp.1464–1480.

De Koning, a P.J. et al., 2011. Repetitive elements may comprise over two-thirds of the human genome. *PLoS genetics*, 7(12), p.e1002384.

Koohy, H., Down, T. a & Hubbard, T.J., 2013. Chromatin Accessibility Data Sets Show Bias Due to Sequence Specificity of the DNase I Enzyme. *PloS one*, 8(7), p.e69853.

Koslicki, D., 2011. Topological Entropy of DNA Sequences. *Bioinformatics*.

Lander, E.S., 2011. Initial impact of the sequencing of the human genome. *Nature*, 470(7333), pp.187–197.

Lander, E.S. et al., 2001. Initial sequencing and analysis of the human genome. *Nature*, 409(6822), pp.860–921.

Lee, T.I. & Young, R.A., 2000. Transcription of eukaryotic protein-coding genes. *Annu Rev Genet*, 34, pp.77–137.

Lenhard, B. & Wasserman, W.W., 2002. TFBS: Computational framework for transcription factor binding site analysis. *Bioinformatics (Oxford, England)*, 18(8), pp.1135–6.

Lifton, R.P. et al., 1978. The Organization of the Histone Genes in Drosophila melanogaster: Functional and Evolutionary Implications. *Cold Spring Harbor Symposia on Quantitative Biology*, 42, pp.1047–1051.

Liu, Z., Venkatesh, S.S. & Maley, C.C., 2008. Sequence space coverage, entropy of genomes and the potential to detect non-human DNA in human samples. *BMC genomics*, 9, p.509.

Mantegna, R.N. et al., 1995. Systematic analysis of coding and noncoding DNA sequences using methods of statistical linguistics. *Physical Review E Statistical Physics Plasmas Fluids And Related Interdisciplinary Topics*, 52(3), pp.2939–2950.

Marco, A. et al., 2009. Relationship between gene co-expression and sharing of transcription factor binding sites in Drosophila melanogaster. *Bioinformatics*, 25(19), pp.2473–2477.

Maston, G. a, Evans, S.K. & Green, M.R., 2006. Transcriptional regulatory elements in the human genome. *Annual review of genomics and human genetics*, 7, pp.29–59.

Mathelier, A. & Wasserman, W.W., 2013. The next generation of transcription factor binding site prediction. *PLoS computational biology*, 9(9), p.e1003214.

Matys, V. et al., 2003. TRANSFAC: transcriptional regulation, from patterns to profiles. *Nucleic Acids Res*, 31(1), pp.374–378.

Mazaheri, P. et al., 2010. Differentiating the protein coding and noncoding RNA segments of DNA using Shannon entropy. *International Journal of Modern Physics C*, 21, pp.1–9.

Miele, A. & Dekker, J., 2008. Long-range chromosomal interactions and gene regulation. *Molecular bioSystems*, 4(11), pp.1046–1057.

Møller, M., 1993. A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. *Neural networks*, 6, pp.525–533.

Montana, D., 1989. Training feedforward neural networks using genetic algorithms. *Proceedings of the eleventh international joint*, pp.762–767.

Montgomery, S.B. et al., 2006. ORegAnno: an open access database and curation system for literature-derived promoters, transcription factor binding sites and regulatory variation. *Bioinformatics*, 22(5), pp.637–640.

Mu, X.J. et al., 2011. Analysis of genomic variation in non-coding elements using population-scale sequencing data from the 1000 Genomes Project. *Nucleic acids research*, 39(16), pp.7058–76.

Murakami, K., Kojima, T. & Sakaki, Y., 2004. Assessment of clusters of transcription factor binding sites in relationship to human promoter, CpG islands and gene expression. *BMC genomics*, 5(1), p.16.

Naughton, B.T. et al., 2006. A graph-based motif detection algorithm models complex nucleotide dependencies in transcription factor binding sites. *Nucleic acids research*, 34(20), pp.5730–9.

Niu, D.-K. & Jiang, L., 2013. Can ENCODE tell us how much junk DNA we carry in our genome? *Biochemical and biophysical research communications*, 430(4), pp.1340–3.

Pallejà, A. et al., 2012. DistiLD Database: diseases and traits in linkage disequilibrium blocks. *Nucleic acids research*, 40(Database issue), pp.D1036–40.

Parkinson, H. et al., 2007. ArrayExpress - a public database of microarray experiments and gene expression profiles. *Nucleic Acids Research*, 35, pp.D747–D750.

Paterson, T. & Law, A., 2012. JEnsembl: a version-aware Java API to Ensembl data systems. *Bioinformatics (Oxford, England)*, 28(21), pp.2724–31.

Plackett, R., 1983. Karl Pearson and the Chi-squared Test. … *Statistical Review/Revue Internationale de Statistique*, 51(1), pp.59–72.

Portales-Casamar, E. et al., 2010. JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles. *Nucleic Acids Research*, 38, pp.D105–D110.

Portales-Casamar, E. et al., 2009. The PAZAR database of gene regulatory information coupled to the ORCA toolkit for the study of regulatory sequences. *Nucleic Acids Res*, 37(Database issue), pp.D54–60.

Prachumwat, A., Devincentis, L. & Palopoli, M.F., 2004. Intron Size Correlates Positively With Recombination Rate in Caenorhabditis elegans. *Genetics*, 1590(March), pp.1585–1590.

Prlić, A. et al., 2012. BioJava: an open-source framework for bioinformatics in 2012. *Bioinformatics (Oxford, England)*, 28(20), pp.2693–5.

Qiu, J., 2006. Epigenetics: unfinished symphony. *Nature*, 441(7090), pp.143–5.

Ranganathan, A., 2004. The Levenberg-Marquardt Algorithm. *Tutoral on LM Algorithm*, (June), pp.1–5.

Riedmiller, M., 1994. Advanced supervised learning in multi-layer perceptrons - From backpropagation to adaptive learning algorithms. *Computer Standards & Interfaces*, 16.

Riedmiller, M. & Braun, H., 1993. A direct adaptive method for faster backpropagation learning: the RPROP algorithm. *IEEE International Conference on Neural Networks*, pp.586–591.

Robertson, E.A. & Zweig, M.H., 1981. Use of receiver operating characteristic curves to evaluate the clinical performance of analytical systems. *Clinical chemistry*, 27(9), pp.1569–1574.

Robertson, G. et al., 2006. cisRED: a database system for genome-scale computational discovery of regulatory elements. *Nucleic Acids Research*, 34, pp.D68–D73.

Romero, E. & Toppo, D., 2007. Comparing Support Vector Machines and Feedforward Neural Networks With Similar Hidden-Layer Weights. *Neural Networks, IEEE Transactions on*, 18(3), pp.959–963.

Sandelin, A. et al., 2004. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic acids research*, 32(Database issue), pp.D91–4.

Sasi, B., Sonawane, P. & Gupta, V., 2013. Coordinated Transcriptional Regulation of Hspa1a Gene by Multiple Transcription Factors: Crucial Roles for HSF-1, NF-Y, NF-ϰB, and CREB. *Journal of molecular …*, pp.1–20.

Saxonov, S., Berg, P. & Brutlag, D.L., 2006. A genome-wide analysis of CpG dinucleotides in the human genome distinguishes two distinct classes of promoters. *Proc Natl Acad Sci U S A*, 103(5), pp.1412–1417.

Schiffmann, W., Joost, M. & Werner, R., 1993. Comparison of Optimized Backpropagation Algorithms. *ESANN*.

Schneider, T., 2010. A brief review of molecular information theory. *Nano communication networks*, 1(3), pp.173–180.

Schones, D.E. et al., 2008. Dynamic regulation of nucleosome positioning in the human genome. *Cell*, 132(5), pp.887–898.

Segal, E. et al., 2006. A genomic code for nucleosome positioning. *Nature*, 442(7104), pp.772–778.

Shannon, C.E., 1949. The mathematical theory of communication. 1963. *M.D. computing : computers in medical practice*, 14(4), pp.306–17.

Shavor, Sherry, et al., 2003. *The Java Developer's Guide to Eclipse*,

Sing, T. et al., 2005. ROCR: visualizing classifier performance in R. *Bioinformatics (Oxford, England)*, 21(20), pp.3940–1.

Smedley, D. et al., 2009. BioMart--biological queries made easy. *BMC genomics*, 10(1), p.22.

Smith, J.T., 2006. Neural Network Verification. In *Methods and Procedures for the Verification and Validation of Artificial Neural Networks*. Springer, pp. 109–161.

Stanley, H.E. et al., 1999. Scaling features of noncoding DNA. *Physica A*, 273(1-2), pp.1–18.

Stormo, G.D., 2000. DNA binding sites: representation and discovery. *Bioinformatics (Oxford, England)*, 16(1), pp.16–23.

Team, R.D.C., 2005. R: A language and environment for statistical computing.

Tetko, I. V, Livingstone, D.J. & Luik, A.I., 1995. Neural network studies. 1. Comparison of overfitting and overtraining. *Journal of chemical information and computer sciences*, 35(5), pp.826–833.

Thomas, D.J. et al., 2007. The ENCODE project at UC Santa Cruz. *Nucleic Acids Research*, 35, pp.D663–D667.

Tillo, D. et al., 2010. High Nucleosome Occupancy Is Encoded at Human Regulatory Sequences. *PLoS One*, 5(2), p.-.

Tomovic, A. & Oakeley, E.J., 2007. Position dependencies in transcription factor binding sites. *Bioinformatics (Oxford, England)*, 23(8), pp.933–41.

Tost, J., 2009. DNA methylation: an introduction to the biology and the disease-associated changes of a promising biomarker. *Methods Mol Biol*, 507, pp.3–20.

Troyanskaya, O.G. et al., 2002. genomic sequences : A fast algorithm for. *Science*, 18(5), pp.679–688.

Tuerk & Gold, 1990. SELEX.

Vaquerizas, J.M. et al., 2009. A census of human transcription factors: function, expression and evolution. *Nature Reviews Genetics*, 10(4), pp.252–263.

Veerla, S., Ringner, M. & Hoglund, M., 2010. Genome-wide transcription factor binding site/promoter databases for the analysis of gene sets and co-occurrence of transcription factor binding motifs. *BMC Genomics*, 11, p.145.

Venter, J.C. et al., 2001. The sequence of the human genome. *Science (New York, N.Y.)*, 291(5507), pp.1304–51.

Wasserman, W.W. & Sandelin, A., 2004. Applied bioinformatics for the identification of regulatory elements. *Nature Reviews Genetics*, 5(4), pp.276–287.

Waterston, R.H. et al., 2002. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420(6915), pp.520–62.

Whitley, D., 1994. A genetic algorithm tutorial. *Statistics and Computing*, 4(2).

Wilming, L.G. et al., 2008. The vertebrate genome annotation (Vega) database. *Nucleic acids research*, 36(Database issue), pp.D753–60.

Xi, L. et al., 2010. Predicting nucleosome positioning using a duration Hidden Markov Model. *BMC Bioinformatics*, 11, p.346.

Xie, X., Rigor, P. & Baldi, P., 2009. MotifMap: a human genome-wide map of candidate regulatory motif sites. *Bioinformatics (Oxford, England)*, 25(2), pp.167–74.

Yuan, G.-C. & Liu, J.S., 2008. Genomic sequence is highly predictive of local nucleosome depletion. *PLoS computational biology*, 4(1), p.e13.

Zhang, Y. et al., 2010. HHMD: the human histone modification database. *Nucleic Acids Research*, 38, pp.D149–D154.

Zhao, Z. & Han, L., 2009. CpG islands: algorithms and applications in methylation studies. *Biochem Biophys Res Commun*, 382(4), pp.643–645.

Appendix

A selection of the main programs and classes created in the production of the modelling system.

i) Data Extraction: Code for data extraction from sources and population of database tables.

Code/Class	**Detail**
Core.pl	Extract Genes and coordinates from Ensembl
CoreGetIntronsExons.pl	Get sequences of Introns/Exons from Ensembl
CorePopulateDB.pl	Extract Flanking sequences from Ensembl
CorePopulateDBTranscripts.pl	Extract details of all transcripts within all genes
CorePopulateGapDB.pl	Extract permitted genes for Entropy modelling
funcGenAll.pl	Extract functional annotations from Ensembl
GetCpGIslands.pl	Extract Ensembl calculation of CpG Island and their location
getGOTerms.pl	XML Extract of Gene Ontology
Oreganno.pl	XML Extraction from ORegAnno
oreganno_Pazar.pl	Extraction for ORegAnno data from Pazar feed
BioMartFetchSeq.java	Http extraction of raw sequences

ii) Data Creation: Code for processing and creating analysable data from raw inputs.

Code/Class	**Detail**
AddEpiData.java	Creates variables from raw epigenetic data
AddExpVer.java	Matches ENCODE with predicted TFBSs
AddNuPop.java	Reads NuPop results from R output and creates table
AggregateGOSlim.java	Groups gene ontology data
NuclesomeOccupancy.java	Selects data, calls nupop.R and reads back results
nupop.R	Supplied code for calculating nupop values.
TFSBReport.java	Creates all potential TFBSs from input sequences

iii) Modelling: Code for the production and verification of the modelling.

Code/Class	**Detail**
Proj1.java	Controls all aspects of modelling, reading data, create network and performs the training
PrintWriter.java	Controls reporting and reading and writing of serialised files
ProjValidator.java	Reads models, extracts new random dataset and performs validation
Bootstrap.java	Bootstrapping analysis for CRMs
CalculateEntropy.java	Calculates entropy for variable length sequences
EBVPeaks.java	Searches for peaks and specific sequences in EBV Genome
FeedForwardNetwork.java	Controls the different layers of the model
FeedForwardLayer.java	A single layer that looks after scoring and activation functions
Train.java	An interface that handles the different types of backpropagation models
GeneticAlgorithm.java	A controlling class for genetic algorithms
Chromosome.java	A class for genetic algorithms that handles the production of the next generation via mating, crossing over and mutations

iv) Utilities: General utilities required in initial testing and modelling stages.

Code/Class	**Detail**
BioView.java	Collection of utilities used in test phase of project, report production etc.
Block2Gene.java	Allocates a LD block to any gene
Butils.java	Various utilities, codon2AA, read and process FASTA files etc.
CombinedReport.java	Initial reporting utilities
CpGIslands.java	Calculation of CpG islands, for comparison with Ensembl extracts
DBConnect.java	Utilities for database processing
EBISoap.java	Uses SOAP to extract gene records one at a time
Fasta.java	Utilities to handle the FASTA format
GenBank2DB.java	Utilities to handle the GENBank format
HotColdMap.java	Produce a heatmap of TFBS predictions
NormaliseAll.java	Takes a dataset and normalises all numeric values
PWM.java	Class for processing PWMs
TFBS.java	Create potential TFBSs from PWMs from JASPAR and TRANSFAC